

# Improving Ontology Recommendation and Reuse in WebCORE by Collaborative Assessments

Iván Cantador, Miriam Fernández, Pablo Castells

Escuela Politécnica Superior  
Universidad Autónoma de Madrid  
Campus de Cantoblanco, 28049, Madrid, Spain

{ivan.cantador, miriam.fernandez, pablo.castells}@uam.es

## ABSTRACT

In this work, we present an extension of CORE [2], a tool for Collaborative Ontology Reuse and Evaluation. The system receives an informal description of a specific semantic domain and determines which ontologies from a repository are the most appropriate to describe the given domain. For this task, the environment is divided into three modules. The first component receives the problem description as a set of terms, and allows the user to refine and enlarge it using WordNet. The second module applies multiple automatic criteria to evaluate the ontologies of the repository, and determines which ones fit best the problem description. A ranked list of ontologies is returned for each criterion, and the lists are combined by means of rank fusion techniques. Finally, the third component uses manual user evaluations in order to incorporate a human, collaborative assessment of the ontologies.

## Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – *information filtering, retrieval models, selection process.*

## General Terms

Algorithms, Measurement, Human Factors.

## Keywords

Ontology evaluation, ontology reuse, collaborative filtering.

## 1. INTRODUCTION

The Web can be considered as a live entity that grows and evolves fast over time. The amount of content stored and shared on the web is increasing quickly and continuously. The global body of multimedia resources on the Internet is undergoing a significant growth, reaching a presence comparable to that of traditional text contents. The consequences of this enlargement result in well known difficulties and problems, such as finding and properly managing all the existing amount of sparse information.

To overcome these limitations the so-called “Semantic Web” trend has emerged with the aim of helping machines to process information, enabling browsers or other software agents to automatically find, share and combine information in consistent ways. At the core of these new technologies, ontologies are envisioned as key elements to represent knowledge that can be understood, used and shared among distributed applications and machines. However, ontological knowledge mining and development are difficult and costly tasks that require major engineering efforts. In this context, ontology reuse becomes an essential need in order to exploit past and current efforts and

achievements. Novel tools have been recently developed, such as ontology search engines [6] represent an important first step towards automatically assessing and retrieving ontologies which satisfy user queries and requests. However, ontology reuse demands additional efforts to address special needs and requirements from ontology engineers and practitioners. It is necessary to evaluate and measure specific ontology features, such as lexical vocabulary, relations [3], restrictions, consistency, correctness, etc., before making an adequate selection. Some of these features can be measured automatically, but others require a human judgment to be assessed.

The Web 2.0 is arising as a new trend where people collaborate and share their knowledge to successfully achieve their goals. Following this aspiration, the aim of this research is to enhance ontology retrieval and recommendation, combining automatic evaluation techniques with explicit users’ opinions and experiences. This work follows a previous approach for Collaborative Ontology Reuse and Evaluation over controlled repositories, named CORE [2]. The tool has been enhanced and adapted to the Web. Novel technologies, such as AJAX<sup>1</sup>, have been incorporated to the system for the design and implementation of the user interface. It has also been improved to overcome previous limitations, such as handling large numbers of ontologies. The collaborative capabilities have also been extended.

## 2. SYSTEM ARCHITECTURE

WebCORE is a web application for Collaborative Ontology Reuse and Evaluation. A user logs into the system via a web browser, and, thanks to AJAX technology and the Google Web Toolkit<sup>2</sup>, dynamically describes a problem domain, searches for ontologies related to this domain, obtains relevant ontologies ranked by several lexical, taxonomic and collaborative criteria, and evaluates by himself those ontologies that he likes or dislikes most.

In this section, we describe the server-side architecture of WebCORE. Figure 1 shows an overview of the system. We distinguish three different modules. The first one, the left module, receives the problem description (Golden Standard) as a full text or as a set of initial terms, than can be extended by the user using WordNet [4]. The second one, represented in the centre of the figure, allows the user to select a set of ontology evaluation techniques to recover the ontologies closest to the given Golden Standard. Finally, the third one, on the right of the figure, is a collaborative module that re-ranks the list of recovered ontologies, taking into consideration previous evaluations of the users.

<sup>1</sup> Garrett, J. J. (2005). *AJAX: A New Approach to Web Applications*. In <http://www.adaptivepath.com/>

<sup>2</sup> Google Web Toolkit, <http://code.google.com/webtoolkit/>

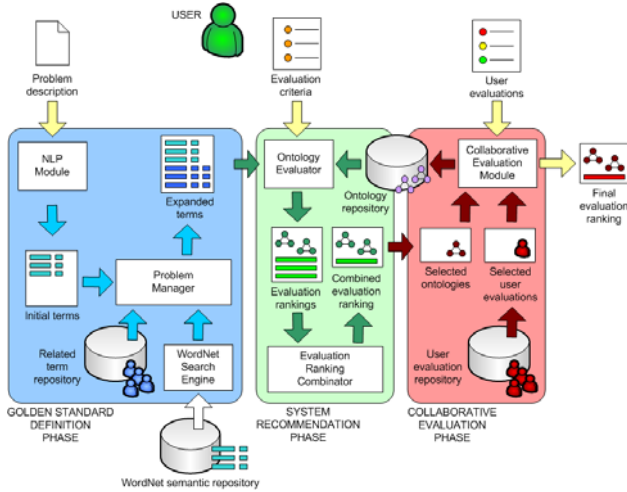


Figure 1. WebCORE architecture

## 2.1 Golden Standard Definition

The first phase of our ontology recommender system is the Golden Standard definition. The user describes a domain of interest specifying a set of relevant terms that will be searched through the concepts (classes or instances) of the ontologies stored in the system. These terms can automatically be obtained by the internal Natural Language Processing (NLP) module, which uses a repository of documents related to the specific domain in which the user is interested in. This NLP module accesses to the repository of documents, and returns a list of pairs (lexical entry, part of speech) that roughly represents the domain of the problem. On the other hand, the list of initial (root) terms can be manually specified. The module also allows the user to expand the root terms using WordNet [4] and some of the relations it provides: hypernym, hyponym and synonym. The new terms added to the Golden Standard using these relations might also be extended again, and new terms can iteratively be added to the problem definition.

The final representation of the Golden Standard is defined as a set of terms  $T(L^G, POS, L^{GP}, R, Z)$  where:

- $L^G$  is the set of lexical entries defined for the Golden Standard.
- POS corresponds to the different Parts Of Speech considered by WordNet: *noun*, *adjective*, *verb* and *adverb*.
- $L^{GP}$  is the set of lexical entries of the Golden Standard that have been extended.
- $R$  is the set of relations between terms of the Golden Standard: *synonym*, *hypernym*, *hyponym* and *root* (if a term has not been obtained by expansion, but is one of the initial terms).
- $Z$  is an integer number that represents the depth or distance of a term to the root term from which it has been derived.

*Example:*  $T_1 = ("genetics", NOUN, "", ROOT, 0)$ .  $T_1$  is one of the root terms of the Golden Standard. The lexical entry that it represents is "genetics", its part of speech is "noun", it has not been expanded from any other term so its lexical parent is the empty string, its relation is "root", and its depth is 0.

Figure 2 shows the interface of the Golden Standard Definition phase. In the left side of the screen, the current list of root terms is shown. The user can manually insert new root terms to this list giving their lexical entries and selecting their parts of speech. Adding new terms, the final Golden Standard definition is

immediately updated: the final list of (root and expanded) terms that represent the domain of the problem is shown in the bottom of the figure. The user can also make term expansion using WordNet. He selects one of the terms from the Golden Standard definition and the system shows him all its meanings contained in WordNet (top of the figure). After he has chosen one of them, the system presents him three different lists with the synonyms, hyponyms and hypernyms of the term. The user can then select one or more elements of these lists and add them to the expanded term list. For each expansion, the depth of the new term is increased by one unit.

In the problem definition phase a collaborative component has been added to the system (right side of Figure 2). This component reads the term currently selected by the user, and searches for all the stored problem definitions that contain it. For each of these problem definitions, the rest of their terms and the number of problems in which they appear are retrieved and shown in the web browser. With this simple strategy the user is suggested the most popular terms, fact that could help him to better describe the domain in which he is interested in.

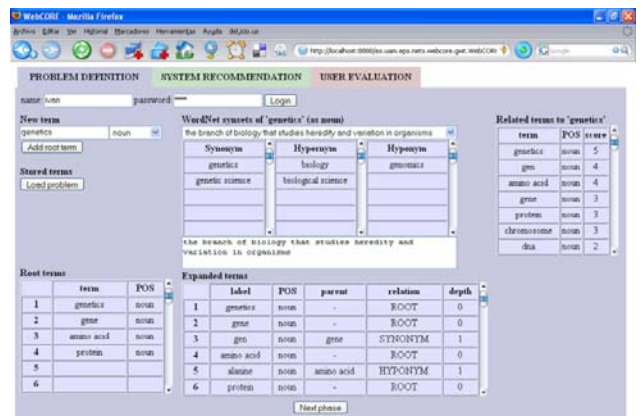


Figure 2. WebCORE problem definition phase

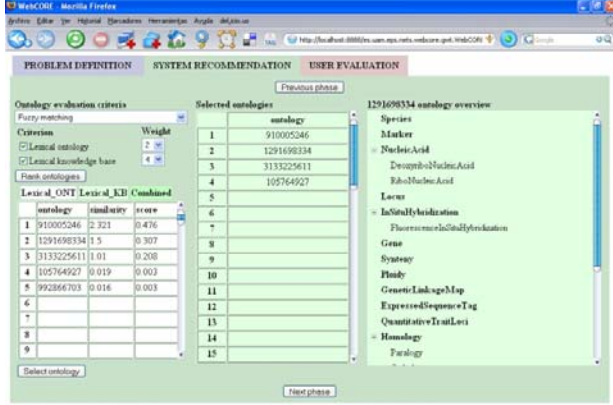
## 2.2 Automatic Ontology Recommendation

Once the user has selected the most appropriate set of terms to describe the problem domain, the tool performs the processes of ontology retrieval and ranking. Our approach to ontology retrieval can be seen as an evolution of classic keyword-based retrieval techniques [5], where textual documents are replaced by ontologies.

The queries supported by our model are expressed using the terms selected during the Golden Standard definition phase. In classic keyword-based vector-space models for information retrieval, each query keyword is assigned a weight that represents the importance of the concept in the information need expressed by the query. Analogously, in our system, the terms included in the Golden Standard are weighted, using the depth measure to indicate the relative interest of the user for each of the terms to be explicitly mentioned in the ontologies.

To carry out the retrieval process, we focus on the lexical level, recovering those ontologies that contain a subset of the terms expressed by the user during the Golden Standard definition. To compute the term matching, two different options are available within the tool: search for exact matches or search for matches based on the *Levenshtein* distance between two terms. Furthermore, the tool also offers two different search spaces, the ontologies and the corresponding knowledge bases.

Figure 3 shows the system recommendation interface. At the left side the user can select the matching methodology (fuzzy or exact), the search spaces (ontology entities and knowledge base entities), and the weight or importance given to each of the previously selected search spaces. In the right part the user can visualize the ontology and navigate across it. Finally, the middle of the interface presents the list of ontologies selected for the user to be evaluated during the collaborative evaluation phase.



**Figure 3.** WebCORE system recommendation phase

Let  $T$  be the set of all terms defined in the Golden Standard definition phase. Let  $d_i$  be the depth measure associate with each term  $t_i \in T$ . Let  $q$  be query vector extracted from the Golden Standard definition, and let  $w_i$  be the weight associated to each of these terms, where for each  $t_i \in T$ ,  $w_i \in [0,1]$ . Then, the weight  $w_i$  is calculated as:

$$w_i = \frac{1}{d_i + 1}$$

This measure gives more relevance to the terms explicitly expressed by the user, and less importance to those ones extended or derived from previously selected terms. An interesting future work could be to enhance and refine the query, e.g. based on terms popularity, or other more complex strategies as terms frequency analysis.

The search engine computes a semantic similarity value between the query and each ontology as follows. We represent each ontology with a vector  $o_j \in O$ , where  $o_{ji}$  is the mean of the term  $t_i$  similarities with all the matched entities in the ontology if any matching exists, and zero otherwise. The components  $o_{ji}$  are calculated as:

$$o_{ji} = \frac{\sum_{m_i \in M_{ji}} w(m_i)}{|M_{ji}|}$$

where  $M_{ji}$  is the set of matches of the term  $t_i$  in the ontology  $o_j$ ,  $w(m_i)$  represents the similarities between the term  $t_i$  and the entities of the ontology  $o_j$  that matches with it,  $M_i$  is the set of matches of the term  $t_i$  within all the ontologies and  $w(m_i)$  represents the weights of each of these matches.

For example, if we define in the Golden Standard a term “acid”, this term may return several matches in the same ontology with different entities as: “acid”, “amino acid”, etc. In order to establish the appropriate weight in the ontology vector,  $o_{ij}$ , the goal is to compute the number of matches of one term in the whole repository of ontologies and give more relevance to those ontologies that have matched that specific term more times.

Each component  $o_{ij}$  contains specific information about the similarity between the ontology and the corresponding term  $t_i$ . To compute the final similarity between the query vector  $q$  and the ontology vector  $o_j$ , the vectorial model calculates the cosine measure between both vectors. However, if we follow the traditional model, we will only be considering the difference between the query and the ontology vectors according to the angle they form, but not taking into account their dimensions. To overcome this limitation, the cosine measure has been replaced by the simple dot product. Hence, the similarity measure between an ontology  $o_j$  and the query  $q$  is simply compute as follows:

$$sim(q, o_j) = q \cdot o_j$$

If the knowledge in the ontology is incomplete, the ontology ranking algorithm performs very poorly. Queries will return less results than expected, the relevant ontologies will not be retrieved, or will get a much lower similarity value than it should. For instance, if there are ontologies about “restaurants”, and “dishes” are expressed as instances in the corresponding Knowledge Base (KB), a user searching for ontologies in this domain may be also interested in the instances and literals contained in the KB. To cope with this issue, our ranking model combines the similarity obtained from the terms that belong to the ontology with the similarity obtained from the terms that belong to the KB using the adaptation of the vector space model explained before. The user can select a value  $v_i \in [1, 5]$  for each kind of search, and this value is then mapped to a corresponding value  $s_i = \frac{v_i}{5}$ . Following this idea, the final score is computed as:

$$s_o \times sim(q, o) + s_{kb} \times sim(q, kb)$$

### 2.3 Collaborative Ontology Evaluation

The third and last phase of the system is compound of a novel ontology recommendation algorithm that exploits the advantages of Collaborative Filtering [1], exploring the manual evaluations stored in the system to rank the set of ontologies that best fulfils the user’s interests.

In WebCORE, user evaluations are represented as a set of five different criteria and their respective values, manually determined by the users who made the evaluations: *correctness*, *readability*, *flexibility*, *level of formality* and *type of model*.

The above criteria can have discrete numeric or non-numeric values. The user’s interests are expressed like a subset of these criteria, and their respective values, meaning thresholds or restrictions to be satisfied by user evaluations. Thus, a numeric criterion will be satisfied if an evaluation value is equal or greater than that expressed by its interest threshold, while a non-numeric criterion will be satisfied only when the evaluation is exactly the given threshold (i.e. in a Boolean or yes/no manner).

According to both types of user evaluation and interest criteria, numeric and Boolean, the recommendation algorithm will measure the degree in which each user restriction is satisfied by the evaluations, and will recommend a ranked ontology list according to similarity measures between the thresholds and the collaborative evaluations.

Figure 4 shows all the previous definitions and ideas, locating them in the graphical interface of the system. On the left side of the screen, the user introduces the thresholds for the recommendations and obtains the final collaborative ontology ranking. On the right side, the user adds new evaluations for the ontologies and checks evaluations given by the rest of the users.

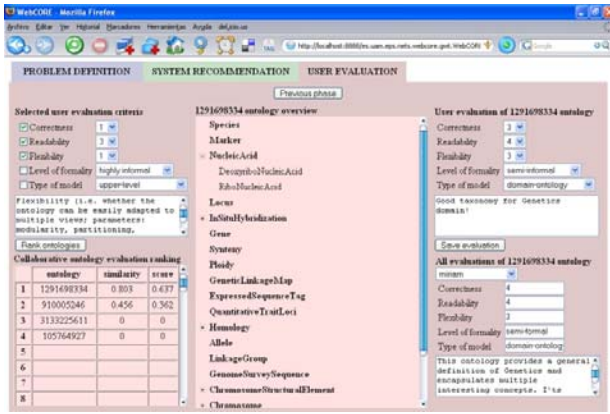


Figure 4. WebCORE user evaluation phase

### 3. EXPERIMENTS

In this section, we present some early experiments that attempt to measure: a) the gain of efficiency and effectiveness, and the b) increment of users' satisfaction obtained with the use of our system when searching ontologies within a specific domain.

The scenario of the experiments was the following. A repository of thirty ontologies was considered and eighteen subjects participated in the evaluations. They were Computer Science Ph.D. students of our department, all of them with some expertise in modeling and exploitation of ontologies. They were asked to search and evaluate ontologies with WebCORE in three different tasks. For each task and each student, one of the following problem domains was selected *family*, *genetics* and *restaurant*.

In the repository, there were six different ontologies related to each of the above domains, and twelve ontologies describing other no related knowledge areas. No information about the domains and the existent ontologies was given to the students.

Tasks 1 and 2 were performed first without the help of the collaborative modules of the system, i.e., the term recommender of the problem definition phase and the collaborative ranking of the user evaluation phase. After all users finished the previous ontology searches and evaluations, task 3 was done with the collaborative components activated. For each task and each student, we measured the time expended, and the number of ontologies retrieved and selected ('reused'). We also asked the users about their satisfaction (in a 1-5 rating scale) about each of the selected ontologies and the collaborative modules.

Tables 1 and 2 contain a summary of the obtained results. Note that measures of task 1 are not shown. We have decided not to consider them for evaluation purposes because we discern the first task as a learning process of the use of the tool, and its time executions and number of selected ontologies as skewed no objective measures.

To evaluate the enhancements in terms of efficiency and effectiveness, we present in Table 1 the average number of reused ontologies and the average execution times for task 2 and 3. The results show a significant improvement when the collaborative modules of the system were activated. In all the cases, the students made use of the terms and evaluations suggested by others, accelerating the processes of problem definition and relevant ontology retrieval.

Table 1. Average number of reused ontologies and execution times (in minutes) for tasks 2 and 3

	Task 2 (without collaborative modules)	Task 3 (with collaborative modules)	% improvement
# reused ontologies	3.45	4.35	26.08
execution time	9.3	7.1	23.8

On the other hand, table 2 shows the average degrees of satisfaction revealed by the users about the retrieved ontologies and the collaborative modules. Again, the results evidence positive applications of our approach.

Table 2. Average satisfactions values (1-5 rating scale) for ontologies reused in tasks 2 and 3, collaborative recommendations and rankings

Task 2	Task 3	% improvement	Initial term recommendation	Final ontology ranking
3.34	3.56	6.58	4.7	4.4

### 4. CONCLUSIONS AND FUTURE WORK

In this paper, a web application for ontology evaluation and reuse has been presented. The novel aspects of our proposal include the use of WordNet to help users to define the Golden Standard; a new ontology retrieval technique based on traditional Information Retrieval models; rank fusion techniques to combine different ontology evaluation measures; and two collaborative modules: one that suggests the most popular terms for a given domain, and one that recommends lists of ontologies with a multi-criteria strategy that takes into account user opinions about ontology features that can only be assessed by humans.

### 5. ACKNOWLEDGMENTS

This research was supported by the Spanish Ministry of Science and Education (TIN2005-06885 and FPU program).

### 6. REFERENCES

- [1] Adomavicius, G., and Tuzhilin, A.: *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*. IEEE Transactions on Knowledge and Data Engineering 17(6): 734-749, 2005.
- [2] Fernández, M., Cantador, I., and Castells, P. *CORE: A Tool for Collaborative Ontology Reuse and Evaluation*. Proceedings of the 4th Int. Workshop on Evaluation of Ontologies for the Web (EON'06), at the 15th Int. World Wide Web Conference (WWW'06). Edinburgh, UK, 2006.
- [3] Maedche, A., and Staab, S.: *Measuring similarity between ontologies*. Proceedings of the 13th European Conference on Knowledge Acquisition and Management (EKAW 2002). Madrid, Spain, 2002.
- [4] Miller, G. A.: *WordNet: A lexical database for English*. *New horizons in commercial and industrial Artificial Intelligence*. Communications of the Association for Computing Machinery, 38(11): 39-41, 1995.
- [5] Salton, G., and McGill, M.: *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.
- [6] *Swoogle - Semantic Web Search Engine*. <http://swoogle.umbc.edu>