Ontology Search Engines

Overview and recommendations

Isabel Azevedo, Carlos Vaz de Carvalho
School of Engineering and GILT – Graphics, Interaction and Learning Technologies
Polytechnic of Porto
Porto, Portugal
{ifp, cmc}@isep.ipp.pt

Eurico Carrapatoso
Faculty of Engineering
University of Porto
Porto, Portugal
emc@fe.up.pt

***Abstract*—The ability to find ontologies is a matter that has been receiving great attention each year, as it is time expensive to develop an ontology from the very beginning without using any work done earlier. In fact, this can be undesirable as many ontologies have been developed and their quality has been assured by different teams. However, currently ontology search engines need to be improved in order to incorporate other functionalities that are not common. This paper analyses tools that make easier the discovery of ontologies that cover some concepts, also providing some recommendations to facilitate the whole process.**

Keywords-ontology; repositories; search engines, Semantic Web

I. Introduction

Ontologies have been increasingly used in the context of the Semantic Web and they have been applied in different areas and projects. On the other hand, the reuse of ontologies is a step that has been proposed in many methodologies for ontology development [1].

Ontology search engine is a tool that does not require an active action from ontology developers, as it automatically searches for and indexes the ontologies they discover. Some examples are Swoogle [2], Watson [3], Sindice [4] and Falcons [5] . They vary in the metadata provided for each ontology, as there is no standard for ontology metadata and exchange.

This work began as part of a broader one that aimed at the development of an ontology reuse module that was incorporated in a repository of educational resources [6, 7] to improve their characterisation and findability, using semantics throughout these processes.

In this paper, ontology search engines are analyzed as tools that help users in the selection of useful ontologies, which are always dependent on the particular application that is envisaged. Thus, evaluation of ontologies in order to identify the suitable ones is out of the scope of this work. The study focuses in a number of aspects, and many of them are not semantic issues, but affect their usefulness.

The rest of the paper is organized as follow. In section 2 the theme of ontology search engines is expanded, and three of them are analysed. In the third section, they are compared through the results returned for some queries, exploring their similarities, but also some differences. In this section, the results are analyzed, substantiating some suggestions. Finally, the last section provides some concluding remarks and general recommendations for the improvement of Semantic Web Search Engines.

II. Search engines analysis

Ontology search engines accept queries in a format that varies from one tool to another. They usually provide results in an XML file. Their broader designation is Semantic Web Search Engines (SWSE), as they provide Semantic Web documents (SWD). However, this latter designation applies to a range of documents, besides ontologies, that fall into two categories: pure SWDs (PSWDs), and embedded SWDs (ESWDs), such as HTML documents with their associated metadata [8].

Different from other types of platforms that can be used to find suitable ontologies, such as ontology repositories, which sometimes only provide browse functionalities, ontology search engines permit a greater degree of automation.

The great amount of results provided by some SWSEs, which do not have concept or ontology search functionalities, disregard their consideration for ontology reuse based on concepts. For instance, a query on Sindice with the term 'Table' returns more than 800,000 results, much more than those returned by other SWSEs (see Table 1). However, a great part of them are not ontologies.

From the list previously mentioned, the more ontology-based search engines are Swoogle, Watson and Falcons, which are described in sections A, B and C, respectively.

They all allow human submission of Semantic Web documents. Also, their architectures include crawling, indexing and analyzing blocks, which are important components of any SWSE.

There are Swoogle's statistics available at its Web site. It has indexed more than 3,800,000 Semantic Web documents and over 10,000 ontologies. It is mentioned in [9] that 11.7 million well-formed RDF/XML documents were crawled.

A. Swoogle

Swoogle was the first search engine dedicated to online semantic data and it remains one of the most popular SWSE.

 ISBN: 978-1-61208-240-0

Its development was partially supported by DARPA and NSF (National Science Foundation).

The current version of Swoogle is 3.1, which has been available since January 2006.

Swoogle's architecture (see Fig. 1) has four major components:

- The Discovery component – It is responsible for collecting candidate URLs. It caches Semantic Web Documents. Swooglebot is the Swoogle's Semantic Web Crawler that produces new candidates to be considered, but conventional search engines are also used for the same purpose. In addition, there is an option to submit sites and documents to be regarded;
- The Indexing component – It analyses the Semantic Web Documents (SWDs) found by the Discovery component and generates some metadata, which characterises the features associated with individual SWDs and Semantic Web Terms (SWTs), but also the relations among them. These metadata intend to improve searches;
- The Analysis component - It uses the metadata generated by the Indexing component to support ranking mechanisms;
- The Search Services module - It allows Swoogle to be used by agents and humans. It is mainly an interface component.

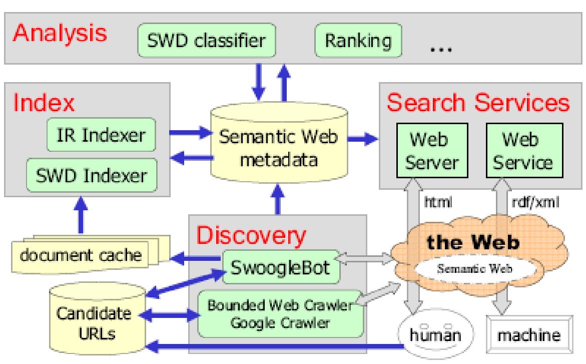

Figure 1. Swoogle 3.1 architecture (from [10]).

The Swoogle ranking method is based on the OntoRank algorithm, which is quite analogous to the PageRank algorithm (used by the Google search engine). Consider a page A which has n pages (T1, T2, ... Tn) with a link to it, as depicted in Fig. 2.

The PageRank of page A can be stated as follows [11]:

$$PR(A) = (1-d) + d\sum_{i=1}^{n}\frac{PR(T_i)}{C(T_i)}$$

In the equation above, d is a normalising factor, whose value can vary from 0 to 1, C(Ti) is defined as the number of links that Ti points to. The PageRank of A (PR(A)) considers the PageRank of each Ti (PR(Ti)). OntoRank adapts the PageRank approach "to expose more ontologies which are important to Semantic Web users" [10], using semantic relations between ontologies. Ding et al. detail the OntoRank method and compare it with the PageRank algorithm.

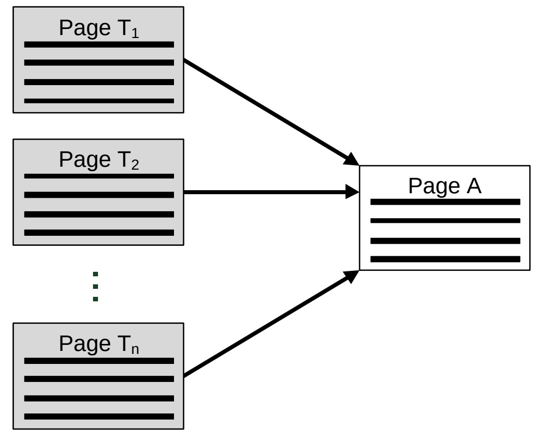

Figure 2. The idea behind PageRank algorithm (adapted from [11]).

It is noteworthy that the keywords specified in a query do not influence the ranking process, just the inclusion of a given document in the results set.

B. *Watson*

The Watson development was partially supported by the NeOn [12] and the OpenKnowledge [13] projects.

The functions (collecting, analysing and querying) of the core components of Watson (see Fig. 3) do not differ significantly from those of Swoogle. These functions correspond to three different layers as follows:

- The ontology crawling and discovery layer is responsible for obtaining semantic data. Any document that cannot be parsed by Jena is disregarded as a way to guarantee that only documents that contain semantic data or ontologies are considered;
- The validation and analysis layer gathers metadata about the semantic data, which is also used for indexing purposes. In addition, semantic relations between ontologies are regarded for the retrieved ontologies (e.g., owl:imports, rdfs:seeAlso, namespaces, derefenceable URIs) in order to detect other sources of ontologies;
- The query and navigation layer is related to the available query interfaces that allow using the Watson functionalities.

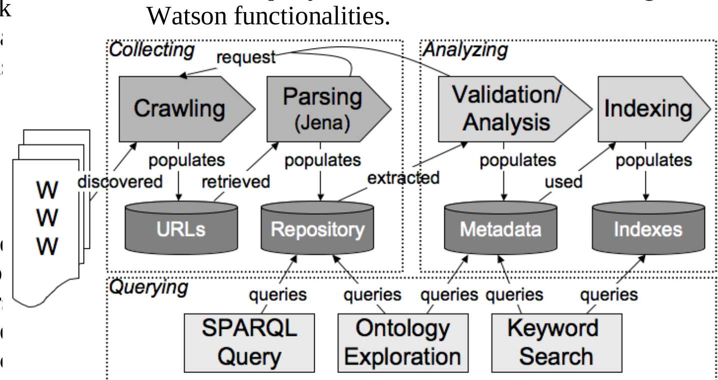

Figure 3. Watson architecture (from [3]).

 ISBN: 978-1-61208-240-0

In [14], it is mentioned that for ranking it is used "an initial set of measures that evaluate ontology complexity and richness". Also, d'Aquin et al. [3] state that "the ranking mechanisms offered by Watson rely on a combination of simple, basic quality measures that are computed in the validation phase and stored along with the ontologies (i.e., structural measures, topic relevance, etc.)". However, the exact ranking method used by Watson is unknown.

A distinctive characteristic of this SWSE in comparison to Swoogle and Falcons is the possibility to review ontologies or see how other users have reviewed it, a trend that have become popular in other areas and that led to the inclusion of user review sections in many different systems. In Watson, that functionally relies on Revyu.com, which is a web site where people can review and rate things.

C. *Falcons*

The Falcons architecture has many components (see Fig. 4). The crawled documents are parsed and the URIs are then processed by the URI repository for further crawling. The quadruple (RDF triple plus the document URI) is stored. These data are processed by the meta-analysis component, which provides detailed ontological information to the metadata module. The indexer updates the next component, which is the basis of the keyword-based search functionalities. Objects are ranked is accordance to their relevance to the query submitted and their popularity. Comprehensive information about all components is provided in [9].

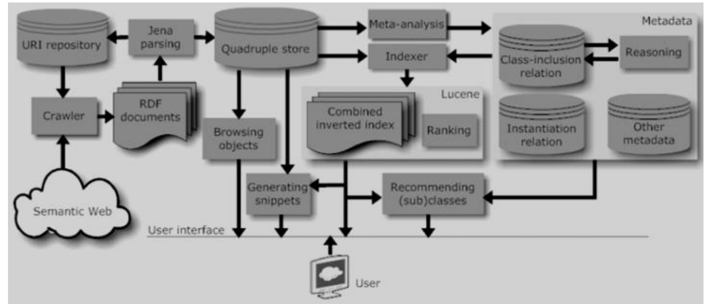

Figure 4. Falcons architecture (from [9]).

Users can use Falcons to search for objects, concepts, ontologies and documents. The object search option is useful when trying to find specific things. Concept search is useful to find classes or properties in ontologies. The option to search ontologies (see Fig. 5) provides a subset of the results returned using the option to search documents, and more metadata fields are considered.

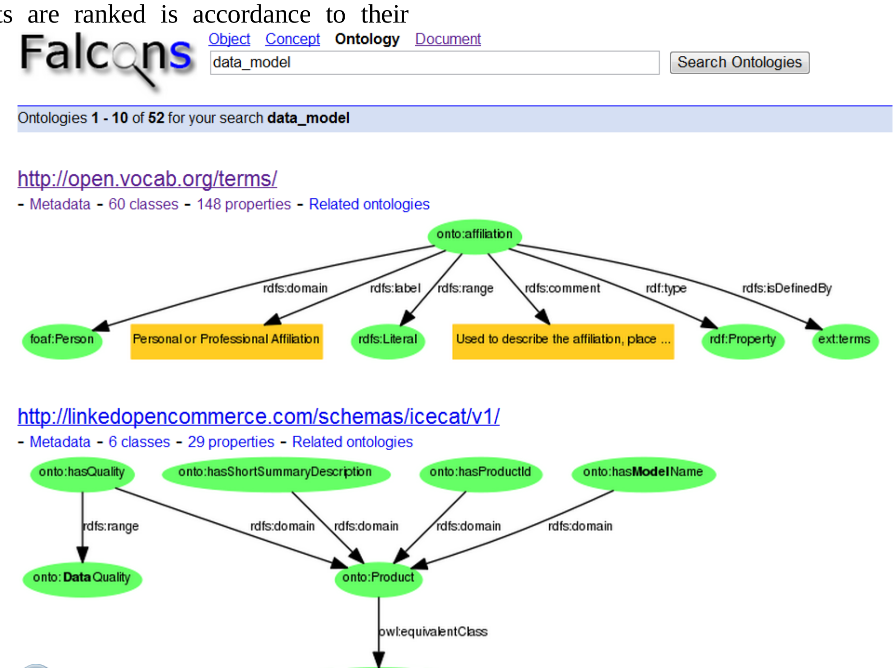

Figure 5. Ontology search with Falcons.

This visual layout of the results provided by Falcons is a distinctive characteristic of this SWSE in comparison to Watson and Swoogle. It lets users understand how the terms are included in each ontology from the results' set.

III. COMPARISON

Table I, Table II and Table III compare Swoogle, Watson and Falcons under the number of results using query terms from different areas of engineering courses. As it was explained, the aim of reusing ontologies in a repository of engineering resources led to this work, and these terms in the tables characterise some engineering courses. The enumeration of the important terms corresponds to one of the recommended steps to follow when developing an ontology [15].

As Falcons does not correctly process the underscore character (see Fig. 5), even if the search strings are put in quotes, terms with this character were not considered in this search engine (in these situations “not applicable” is used in the tables). This point is expanded in the next section.

For the queries submitted to Watson, only classes and properties were considered and local names were regarded. The same options were used at Swoogle (using the `def` specifier). For Falcons the ontology search was used, but it is not possible to select exactly what is of interest, for instance, just classes and/or properties. Thus, the returned ontologies were manually inspected in order to consider just the ontologies fulfilling the same characteristics used in the other search engines.

Table I shows the number of results for some search strings, comparing the results found by Swoogle, Watson and Falcons, but also the number of available results considering only the best ten ranked documents in the results set. Swoogle and Watson do not cope with different writing styles. For instance, the results found for ‘DataModel’ do not include those returned for ‘Data_model’.

Table II provides the results obtained when some concepts from a Statistics course were considered. Table III shows the results found using some concepts from a Chemical course.

TABLE I. NUMBER OF RESULTS FOR SOME DATABASE CONCEPTS EXPRESSED IN DIFFERENT WAYS

Search string	**Swoogle**		**Watson**		**Falcons**	
	Number of results	**Number of available results (Top Ten)**	**Number of results**	**Number of available results (Top Ten)**	**Number of results**	**Number of available results (Top Ten)**
‘Distributed_Database’	0	0	0	0	Not applicable	Not applicable
‘DistributedDatabase’	2	2	0	0	0	0
‘Distributed_Databases’	3	2	0	0	Not applicable	Not applicable
‘DistributedDatabases’	0	0	0	0	0	0
‘Data_model’	13	5	1	0	Not applicable	Not applicable
‘DataModel’	11	7	1	1	0	0
‘DataModels’	0	0	0	0	0	0
‘Data_models’	3	2	1	1	Not applicable	Not applicable
‘Table’	816	6	30	9	25	7
‘Tables’	77	9	4	1	1	1

TABLE II. NUMBER OF RESULTS FOR SOME STATISTICS CONCEPTS EXPRESSED IN DIFFERENT WAYS

Search string	**Swoogle**		**Watson**		**Falcons**	
	Number of results	**Number of available results (Top Ten)**	**Number of results**	**Number of available results (Top Ten)**	**Number of results**	**Number of available results (Top Ten)**
‘Sampling’	225	9	5	3	13	10
‘Samplings’	0	0	0	0	0	0
‘Probability’	232	6	10	6	6	5
‘Probabilities’	2	0	0	0	0	0
‘Linear_regression’	1	1	0	0	Not applicable	Not applicable
‘LinearRegression’	10	2	2	0	0	0
‘LinearRegressions’	0	0	0	0	0	0
‘Linear_regressions’	0	0	0	0	Not applicable	Not applicable
‘Probability_distribution’	1	1	0	0	Not applicable	Not applicable
‘ProbabilityDistribution’	1	0	0	0	0	0
‘ProbabilityDistributions’	0	0	0	0	0	0
‘Probability_distributions’	0	0	0	0	Not applicable	Not applicable

 ISBN: 978-1-61208-240-0

TABLE III. NUMBER OF RESULTS FOR SOME CHEMICAL CONCEPTS EXPRESSED IN DIFFERENT WAYS

Search string	Swoogle		Watson		Falcons	
	Number of results	Number of available results (Top Ten)	Number of results	Number of available results (Top Ten)	Number of results	Number of available results (Top Ten)
'Periodic_table'	0	0	1	0	Not applicable	Not applicable
'PeriodicTable'	1	0	0	0	0	0
'PeriodicTables'	0	0	0	0	0	0
'Periodic_tables'	0	0	0	0	Not applicable	Not applicable
'solution'	400	6	13	9	10	7
'solutions'	34	6	5	1	3	3
'Acid'	621	7	31	5	23	9
'Acids'	50	9	3	1	0	0
'Base'	1,625	4	27	6	79	5
'Bases'	48	5	3	1	0	0

From the experiments here documented it was found that:

- Search strings that can be considered very generic, such as 'Base', 'Table' or 'Solution' return many results. However, a great number of those results are not really for the envisaged area. For example, the results returned for the search string 'Base', included ontologies with classes for baseball, database, and space subjects, among others. Obviously, it does not mean that search engines did not function correctly, but if users can supply many keywords of possible interest (using the OR operator), it might be possible to consider each of them differently at least in the ranking process. Swoogle is the only one to allow the use of the logical operator OR, but each term used does not affect how the others are regarded;
- Although the common conventions of using the singular form in concept names and the CamelCase style to write compound words or phrases, followed by the W3C itself, these are not universally followed. The use of separator (underscore or no character in accordance with CamelCase naming convention) and singular or plural nominal word form in the submission of queries to SWSEs lead to different sets of results, which are not enclosed in the others;
- It was impossible to analyse all the results returned, but generally there is not an overlap in the top ten results provided by Swoogle, Watson and Falcons. It can be a result from the use of different ranking methods, but for some search strings, one SWSE provided no results, while the others returned. Although it is declared in that Watson uses a specialised crawler for Swoogle, it does not seem that it has been active.

Ontology versioning is "the ability to handle changes in ontologies by creating and managing different variants of it", and this subject is deeply analysed in [16]. Although Watson has some version control mechanisms and it is "able to detect some form of duplication of ontologies" [17], the same version of a given ontology can be returned by Watson, or even different versions of the same ontology. For instance, for the search string 'Base' the results returned by Watson include some ontologies that correspond to different versions of the same file.

KANNEL is a framework for detecting and managing ontology relations for large ontology repositories [18]. It was used in conjunction with Watson, with interesting results [19]. It was noticed an improvement in the efficiency of search engines tasks, but also, in the satisfaction of the users involved in these activities. However, the use of KANNEL is not integrated in Watson at this moment.

Version detection problems were also identified in the results provided by Swoogle (see Fig. 6) and Falcons (see Fig. 7).

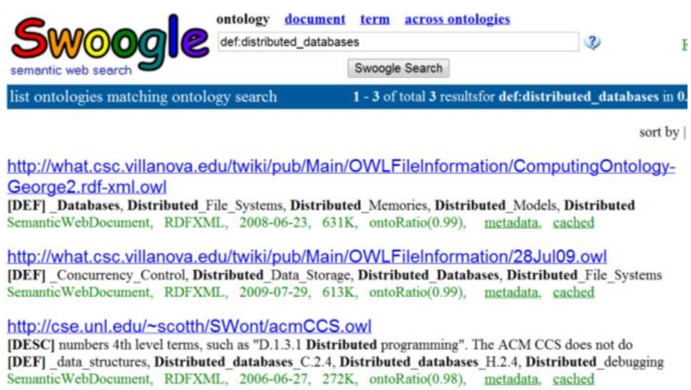

Figure 6. Top results found by Swoogle search using 'distribued_databases' as search string (partial view).

In Fig. 6, the first two ontologies correspond to different versions of the same ontology. The older one appears before and at first perhaps because it had been much more used than the newer one, which affects their ranks. Detection of versioning relationships between documents from the Swoogle's Semantic Web archive was described in [20] and perhaps version control information will start to be considered.

The number of results is not the only criterion to be considered, but it is important as it should be easier to find appropriate ontologies in a large set. However, the results were analysed to determine by sampling if the top ten results were relevant, and they were. For instance, one of the results provided by Swoogle using 'Distributed_Databases' as search string is the computing ontology [21]. However, future studies need to examine the results in details to allow a further comparison at this level.

Some other aspects that were studied were:

- The existence of a limit number of queries accepted;
- The existence of multiple options to sort the results;
- The metadata provided by each returned ontology;
- The possibility of specifying many terms, all to be considered (use of logical operator AND);
- The possibility of specifying many terms to be considered alternatively (use of logical operator OR);
- The ability to dynamically discover semantic data depends on available APIs to access the semantic resources collected by Semantic Web search engines.

These points and others already discussed, as well as some statistical information are summarised in Table IV.

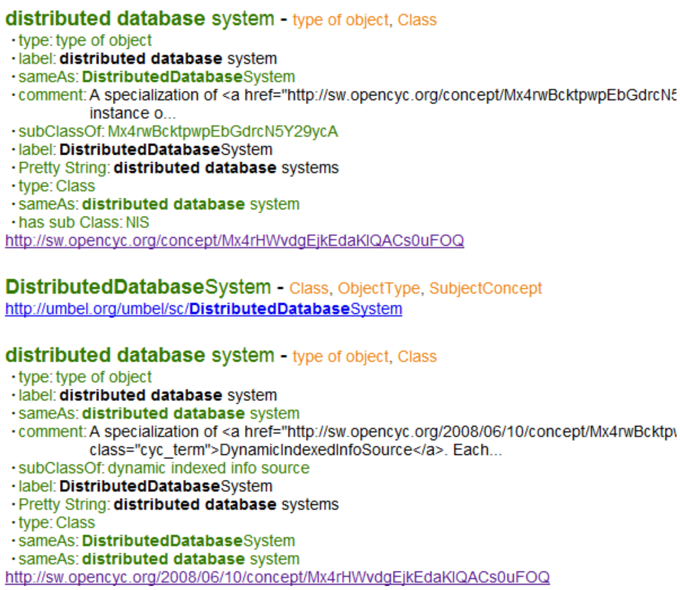

Figure 7. Top results found by Falcons using 'distribued_database' as search string (partial view).

Some common problems were detected. First, in the results set there was a number of ontologies that were no longer available. For instance, one of the three returned ontologies for the query 'distributed_databases' has been unavailable (the first one - see Fig. 6) for more than two years. In that case it is known that this ontology has a newer version (whose URI is http://what.csc.villanova.edu/twiki/pub/Main/OWLFileInformation/28Jul09.owl). Thus, it does not seem that Swoogle has an efficient version control mechanism and, as stated before, Watson suffers from the same problem. However, due to recent versioning developments and experiments that used ontologies indexed by them, it is envisaged the changes will take place soon.

TABLE IV. SWOOGLE, WATSON AND FALCONS – A COMPARISON

Characteristic	Swoogle	Watson	Falcons
Available APIs	Yes	Yes	Yes[1]
Unlimited number of queries	No	Yes	Yes[2]
Multiple sorting possibilities	Yes	No	No
Provision of rich ontology metadata	Yes	Yes	Yes
Use of OR to specify possible terms	Yes	No	No
Use of AND to specify all terms	Yes	Yes	Yes
Possibility to see how other users considered the ontology or rated it	No	Yes	No
Number of crawled SWDs	>3,000,000[3]	-	>11,700,000[4]

[1] The RESTful APIs are described at http://ws.nju.edu.cn/falcons/api/index.jsp, but they were unavailable during the study here documented.
[2] It was not possible to test through the API.
[3] Data from July 2012.
[4] Data from August 2008 [9].

Another point to be improved in SWSEs is the use of wildcards, not their acceptance but how they are treated and the results of their usage. For instance, submitting a query string to Swoogle like "data*model" provides the same results as a query string like 'data_model', and they not include the results obtained with a query string like 'datamodel'. Watson has a similar problem. A query submitted to Watson specifying 'data*model' returns the same results returned by a query like "datamodel", not including the results returned by a query like "data_model". Falcons also accepts wildcards but their effect is the same obtained by the use of the whitespace character or the logical operator AND.

Besides these aspects, the automatic detection of ontology relations, others than versioning, can simplify the results' analysis. For instance, the automatic detection of the inclusion of concepts of one ontology in another one can be a useful functionality, but not yet common. More powerful indexing schemas to deal with similarity and relatedness between concepts at different levels should also be considered. For instance, a search using a query term such as 'relational_model' will fail to provide ontologies with the concept 'relational_data_model', but they have a similarity score near 0.76 using Levenshtein distance [22].

Google Knowledge Graph [23] can be seen as preliminary step in order to provide structured results for keyword-based searches submitted to Google search engine, considering that there are "things, not strings". Such approach in Ontology Search Engines could also help them regard, for instance, that table can be a piece of furniture, but also something meaningful in database area.

IV. CONCLUSION

The comparison of ontology search engines showed that lexical variations, such as the use of separators or not in the query terms and their specification in singular/plural form affect the results. Thus, although SWSEs have to be able to deal with diverse writing styles, currently they are not.

In addition, version control and object coreferencing detection are important for many applications, and also in ontology search engines, as it was discussed in the previous sections. However, at this moment Semantic Web Search Engines do not identify version ontology versions or do not show users this kind of information when they are trying to find ontologies. Changes are expected soon to Swoogle and Watson, as it was discussed.

Finally, a federated query service able to submit queries to multiple sources and a robust but flexible ranking strategy can benefit ontology developers as there is not a considerable overlap among results returned from different ontology search engines.

In addition, the data were collected from August 2010 to July 2011, it will be useful to consider how the results will vary from those reported here in the future, which can provide some insights into the way SWSEs crawl the Web and find new ontologies.

ACKNOWLEDGMENT

This work was supported in part by the PhD grants SFRH/BD/43437/2008 (Portuguese Foundation for Science and Technology) and SFRH/PROTEC/49362/2009 (Ministry of Science, Technology and Higher Education), and the project CASPOE - Characterization of Semantics and Pragmatics of Educational Objects (PTDC/EIA/65387/2006), with funding from the Portuguese Foundation for Science and Technology.

REFERENCES

[1] K. Siorpaes and E. Simperl, "Human intelligence in the process of semantic content creation," World Wide Web, vol. 13, pp. 33–59, 2010.

[2] L. Ding, T. Finin, A. Joshi, R. Pan, R. S. Cost, Y. Peng, P. Reddivari, V. Doshi and J. Sachs, "Swoogle: A search and metadata engine for the semantic web," Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management, 2004.

[3] M. d'Aquin, M. Sabou, M. Dzbor, C. Baldassarre, L. Gridinoc, S. Angeletou and E. Motta, "Watson: A gateway for the semantic web," European Semantic Web Conference (ESWC 2007), Innsbruck, Austria, 2007.

[4] E. Oren, R. Delbru, M. Catasta, R. Cyganiak, H. Stenzhorn and G. Tummarello, "Sindice.Com: A document-oriented lookup index for open linked data," International Journal of Metadata, Semantics and Ontologies, vol. 3, 2008.

[5] G. Cheng, W. Ge and Y. Qu, "Falcons: Searching and browsing entities on the semantic web," Proceedings of the 17th International Conference on World Wide Web, Beijing, China, 2008.

[6] I. Azevedo, R. Seiça, A. Ortiz, E. Carrapatoso and C. V. d. Carvalho, "A semantic approach for learning objects repositories with knowledge reuse," EKAW 2010, Lecture Notes in Artificial Intelligence, Lisbon, Portugal, 2010, pp. 580–589.

[7] I. Azevedo, "Semantic and pragmatic characterisation of learning objects," Informatics. Porto, Portugal, Faculdade de Engenharia da Universidade do Porto. PhD thesis: 317 2012.

[8] L. Ding and T. Finin, "Characterizing the semantic web on the web," The 5th International Semantic Web Conference, Athens, GA, USA, 2006.

[9] G. Cheng and Y. Qu, "Searching linked objects with falcons: Approach, implementation and evaluation," International Journal on Semantic Web and Information Systems, vol. 5, pp. 50-71, July-September 2009 2009.

[10] L. Ding, R. Pan, T. Finin, A. Joshi, Y. Peng and P. Kolari, "Finding and ranking knowledge on the semantic web," The 4th International Semantic Web Conference, Galway, Ireland, 2005.

[11] L. Page, S. Brin, R. Motwani and T. Winograd, "The pagerank citation ranking: Bringing order to the web," November 1999 1999.

[12] "Neon project," http://www.neon-project.org, retrieved: June, 2012.

[13] "Openknowledge project," http://www.openk.org, retrieved: June, 2012.

[14] M. d'Aquin, E. Motta, M. Sabou, S. Angeletou, L. Gridinoc, V. Lopez and D. Guidi, "Toward a new generation of semantic web applications," IEEE Intelligent Systems, vol. 23, pp. 20 - 28, May-June 2008 2008.

[15] N. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology. Technical report ksl-01-05," Stanford Medical Informatics, Stanford 2001.

[16] M. Klein and D. Fenselpages, "Ontology versioning on the semantic web," The International Semantic Web Working Symposium (SWWS), Stanford, USA, 2001.

[17] M. d'Aquin, C. Baldassarre, L. Gridinoc, M. Sabou, S. Angeletou and E. Motta, "Watson: Supporting next generation semantic web applications," WWW/Internet 2007 Conference, Villareal, Spain, 2007.

[18] C. Allocca, "Automatic identification of ontology versions using machine learning techniques," in Lecture notes in computer science. vol. 6643, 2011, pp. 352-366.

[19] C. Allocca, M. d'Aquin and E. Motta, "Impact of using relationships between ontologies to enhance the ontology search results," in Lecture notes in computer science. vol. 7295, 2012, pp. 453-468.

[20] K. Viswanathan and T. Finin, "Text based similarity metrics and delta for semantic web graphs," 9th International Semantic Web Conference (Poster Session), Shanghai, Chima, 2010.

[21] R. Kamali, L. Cassel and R. LeBlanc, "Keeping family of computing related disciplines together," The 5th conference on Information Technology Education, Salt Lake City, UT, USA, 2004, pp. 241 - 243.

[22] V. Levenshtein, "Binary codes for correcting deletions, insertions, and reversals," Doklady Akademii Nauk SSSR, vol. 163, pp. 845–848, 1965.

[23] Google Official Blog, "Introducing the knowledge graph: Things, not strings," http://googleblog.blogspot.pt/2012/05/introducing-knowledge-graph-things-not.html, 2012, retrieved: June, 2012.

