YaSemIR: Yet another Semantic Information Retrieval System

Davide Buscaldi, Haïfa Zargayouna

YaSemIR: Yet Another Semantic Information Retrieval System

Davide Buscaldi[1] and Haïfa Zargayouna[1]

LIPN, Université Paris XIII, F-93430 Villetaneuse
{davide.buscaldi,haifa.zargayouna}@lipn.univ-paris13.fr

Abstract. In this paper we present YaSemIR, a free open-source Semantic Information Retrieval system based on Lucene. It takes one or more ontologies in OWL format and a terminology associated to each ontology in SKOS format to index semantically a text collection. The terminology is used to annotate concepts in documents, while the ontology is used to exploit the taxonomic information in order to expand these with their subsumers. YaSemIR is a flexible system that may be configured to work with different ontologies, on various types of documents.

1 Context

The number of Ontology-based IR systems has been continuously growing in the last years, boosted both by the Semantic Web (SW) and Information Retrieval (IR) research communities. Unfortunately, the proposed solutions are heterogeneous in methods, test collection, scope and standards adopted [11]. Various criteria have been proposed to classify these systems [4]; among these, one is particularly discriminant with respect to the SW or IR nature of a system: whether a system is oriented to data or document retrieval (that is the case of our system).

Independently from the choice of a SW or IR perspective, these systems made some hypothesis: (i) domain ontologies are available, (ii) these ontologies are usable both to annotate documents and to retrieve them.

As the semantic web is continuously expanding and thanks to the effort of standardization, there are more and more ontologies available online. They can be selected using web interfaces such as Watson [1]. The second hypothesis means that the available ontologies are large and explicit enough to cover the vocabulary used in documents. Although we can affirm that the first hypothesis is realistic, the second hypothesis is more difficult to realize in practice. The usability of ontologies is related to their lexical information and to the performance of the annotation process (which links documents to ontologies). Therefore, it is difficult to assess the performance of an Ontology-based IR Systems independently from the coverage allowed by the ontologies it uses.

In literature, the coverage problem is addressed by enriching the knowledge base during the annotation process [6, 9], or proposing to combine both keyword and ontology-based search. Works that propose such a combination can be

[1] `http://watson.kmi.open.ac.uk/WatsonWUI/`

divided in two categories depending on the combination method: composition or merge. [10] propose to apply a spread activation algorithm on a graph build from the documents retrieved by a traditional search engine. [5] propose three strategies that perform a semantic search and reasoning and then rank results with a web search engine. More recent works propose to merge the results of the two searches, [2] propose to return the intersection of sets of documents retrieved by keyword matching and semantic matching (retrieved from an RDF store). Our work is situated closer to the IR perspective and belongs to the last category of hybrid search.

The integration of multiple ontologies is also promising to address the coverage problem, to our knowledge, only PowerAqua system [7] enables to take into account several ontologies. However, PowerAqua is designed with a question answering perspective.

The quantity of models and system proposed in literature is not proportional to the quantity of tools that have been produced and, notably, distributed as free and open source software. Many studies were also tailored for a specific, domain-closed task [8, 2] and it is not clear if the proposed approach would scale. We would like to propose extensions to well-known free, open source IR systems to cope with semantic search. The advantage of such an open source systems is to isolate different semantic components and thus enable to set up comparative evaluation of systems (relying on the same annotation component for example).

In this paper, we introduce an open-source Ontology-based IR system, based on Lucene[2], which takes advantage from standard SW formalisms: OWL and SKOS[3]. YaSemIR is a flexible system that may be configured to work with different ontologies, on various types of documents. A development version is available at the address [4]. A stable, complete and easily configurable version is currently being tested.

2 System Architecture

The key components of YaSemIR are:

- An *annotation* module, which identifies the occurrence of an ontology concept in the documents of the collection or the input query;
- A standard indexing module, based on Lucene. This module creates a standard index, based on Lucene's vector space model implementation;
- A semantic indexing module, which takes the annotations produced by the annotation module, expands them using the ontology and stores the expanded annotations in a separate semantic index (one separate index for each ontology used);
- A ranking module, which takes the scores calculated using keywords and combines them with the scores calculated using the concepts;

[2] `http://lucene.apache.org/`
[3] `http://www.w3.org/2004/02/skos/`
[4] `https://github.com/dbuscaldi/YaSemIR`

– A *knowledge battery* (KB), provided by the user, composed by one or more ontologies (in OWL) and a *terminology* file (in SKOS format) for each ontology, which contains the concept labels, that is, the lexical denotation of concepts: they trigger the detection of a concept in a text. In the case of using multiple ontologies, they could be different conceptualization of the same domain, to improve concept coverage, or refer to different domains, to allow to better represent documents that may pertain to more than one domain.

Let us examine how these components are used in the indexing and the search phase (1).

Indexing After the knowledge battery has been loaded, the first step carried out by YaSemIR is to create a *label index* which maps the stemmed labels into the corresponding concepts. If no terminology were provided, the system would attempt to extract the labels from the concepts URIs of the respective ontology.

After the creation of the label index, the actual indexing is carried out. During this phase, the semantic indexing module parses the documents. Each documents D is passed to the annotation module which fetches the index labels and returns a set of concepts $C_D = \{c_1, \ldots, c_n\}$. This set of document concepts is expanded with their ancestors in the ontology $A_D = \{c_{n+1}, \ldots c_m\}$. Therefore, each document is annotated with the set of concepts $C_D \cup A_D$ which enable to map with general queries' concepts even if they do no occur in the document. The document annotation is stored in the semantic index.

The complete text of the document is indexed with the standard vector model, using the Lucene standard vector model. We chose to maintain a standard, key-word based index in order to overcome the KB coverage problems.

Search The search phase is triggered by the user with a natural language query. This query is analyzed by the annotation module to extract a set of concepts $Q = \{c_1, \ldots, c_k\}$, following the same procedure for the document annotation. This representation is searched in the semantic index, while the base query is searched in the standard index. The keyword-based search relies on Lucene mapping and scoring[5]. The semantic search first returns the list of documents that contain at least one query concept (or a descendant of a query concept). The aim of the semantic score is to return first documents that contain concepts that are the most similar to the query concepts. In our initial evaluation of the system, we used a simple semantic score based on the conceptual similarity score defined by [14]:

$$s(c_i, c_j) = \frac{2 \cdot depth(lca_{ij})}{depth(c_i) + depth(c_j)}. \quad (1)$$

[5] a cosine measure

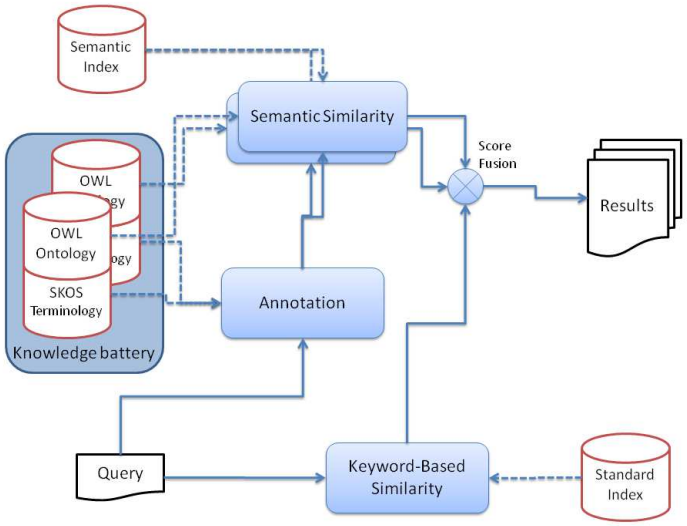

Fig. 1. Overview of the search process.

Other conceptual similarity measures exist in the literature [1] and they can be easily integrated to YaSemIR. The semantic score is calculated as follows:

$$SS_{O_B}(Q,D) = \frac{\sum_{c_i \in Q} w(r(c_i)) \max_{c_j \in C_D} s(c_i,c_j)}{\sum_{c_i \in Q} w(r(c_i))}; \forall c_i, c_j \in O_B \quad (2)$$

Where $r(c_i)$ is the top-level concept (a direct child of an ontology root) for c_i, and $w(r(c_i))$ a weight assigned to model the relative importance of a sub-hierarchy over another. This is useful in order to give more importance to concepts appearing in one ontology than another. It can also be useful if the ontology contains different sub-domains. In YaSemIR, w can be specified in one of the following ways:

- $w(c_i) = 1, \forall c_i \in O_1 \cup O_2 \cup \ldots \cup O_B$ (All concepts in any of the ontologies of the KB have the same weight);
- $w(c_i) = df(c_i)/N$: frequency of concept c_i in collection divided by the number of documents in the collection (frequent concepts are more important);
- $w(c_i) = -\log(df(c_i)/N)$: the inverse document frequency of c_i (rare concepts are more important).

Moreover, a weight can be assigned by the user to each ontology (or to subcategories).

Once a semantic score is calculated for each of the ontologies in the KB $SS_{O_1}(Q,D), \ldots SS_{O_B}(Q,D)$, and the standard score $score_{Luc}(Q,D)$ is obtained from the standard index. These scores are combined into a single score using

the CombANZ strategy [12]. CombANZ is defined as the average of all non-zero scores, while CombMNZ is defined as the sum of all scores, multiplied by the number of non-zero scores. The CombANZ-calculated score is used to rank the documents which are returned to the user. The final score is as follows:

$$score(Q, D) = \frac{\sum_{i \in 1..B} SS_{O_i}(Q, D) + score_{Luc}(Q, D)}{m} \quad (3)$$

Where B is the number of available ontologies and m the number of non-zero scores.

3 First Results and Perspectives

We presented a free, open-source semantic IR system based on Lucene, which exploits concept labels to annotate documents and queries. We carried out a preliminary evaluation on the OHSUMED test collection[6], using the BIKE ontology[7] as KB. We carried out the experiments comparing a keyword-based Lucene baseline, the results obtained with the manual concept annotation included in the collection, and the results obtained with the automatic annotation based only on concept names. The MAP obtained with the manual annotation was 0.297, compared to 0.254 obtained with Lucene (baseline keywords-only) and 0.249 obtained with the automatic annotation. The results showed significant differences in precision depending on errors in the annotation of queries and documents, highlighting the importance of the annotation process in semantic IR. As further work, we plan to integrate state of the art automatic annotation tools when available such as those reported in [13] and to experiment system scalablity by testing it with large knowledge bases such as dbpedia.

We also plan to look for additional ontologies to test the system with multiple ontologies. [3] proposed 40 public ontologies covering a subset of TREC domains (WT10G documents) enriched by available knowledge bases associated with these ontologies. The integration of multiple ontologies proposed by YaSemIR can also be extended in order to work in a multilingual environment, taking advantage from the multilingual features of SKOS.

Acknowledgments

This work has been partially financed by the Labex EFL (ANR/CGI).

References

1. A. Bernstein, E. Kaufmann, C. Kiefer, and C. Burki. Simpack: A generic java library for similarity measures in ontologies., 2005.

[6] `http://ir.ohsu.edu/ohsumed/ohsumed.html`

[7] It is the MeSH ontology in OWL format developed by the Biomedical Knowledge Engineering Laboratory (BIKE[8]) at Seoul National University.

2. R. Bhagdev, S. Chapman, F. Ciravegna, V. Lanfranchi, and D. Petrelli. Hybrid Search: Effectively Combining Keywords and Semantic Searches. In *European Semantic Web Symposium / Conference*, pages 554–568, 2008.
3. M. Fernandez, V. Lopez, E. Motta, M. Sabou, V. Uren, D. Vallet, , and P. Castells. Using trec for cross-comparison between classic ir and ontology-based search models at a web scale. In *Semantic search workshop at 18th International World Wide Web Conference*, 2009.
4. M. Fernndez, I. Cantador, V. Lpez, D. Vallet, P. Castells, and E. Motta. Semantically enhanced information retrieval: An ontology-based approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4):434 – 452, 2011. JWS special issue on Semantic Search.
5. T. Finin, J. Mayfield, A. Joshi, R. Cost, and C. Fink. Information retrieval and the semantic web. In *38th Annual Hawaii International Conference*, 2005.
6. A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff. Semantic annotation, indexing, and retrieval. *Journal of Web Semantics*, 2:49–79, 2004.
7. V. Lopez, M. Fernández, E. Motta, and N. Stieler. Poweraqua: Supporting users in querying and exploring the semantic web. *Semantic Web*, 3(3):249 – 265, 2012.
8. H.-M. Muller, E. E. Kenny, and P. W. Sternberg. Textpresso: An Ontology-Based Information Retrieval and Extraction System for Biological Literature. *Plos Biology*, 2, 2004.
9. A. Reymonet, J. Thomas, and N. Aussenac-Gilles. Ontologies et recherche d'information : une application au diagnostic automobile. In *Acte des 21èmes Journées Francophones d'Ingénierie des Connaissances*, pages 283 – 294. Ecole des Mines d'Alès, 2010.
10. C. Rocha, D. Schwabe, and M. P. Aragao. A hybrid approach for searching in the semantic web. In *Proceedings of the 13th international conference on World Wide Web*, pages 374–383, New York, NY, USA, 2004. ACM.
11. P. Scheir, V. Pammer, and S. N. Lindstaedt. Information retrieval on the semantic web - does it exist? In *Lernen, Wissensentdeckung und Adaptivitt*, pages 252–257, 2007.
12. J. A. Shaw and E. A. Fox. Combination of Multiple Searches. In *Text REtrieval Conference*, 1994.
13. V. Uren, P. Cimiano, J. Iria, S. Handschuh, M. Vargas-Vera, E. Motta, and F. Ciravegna. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(1):14–28, Jan. 2006.
14. Z. Wu and M. S. Palmer. Verb semantics and lexical selection. In *Meeting of the Association for Computational Linguistics*, pages 133–138, 1994.