Yaanii: Effective Keyword Search over Semantic dataset

Roberto De Virgilio
Dipartimento di Informatica e Automazione
Universitá Roma Tre
Rome, Italy
devirgilio@dia.uniroma3.it

Paolo Cappellari
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada
cappellari@ualberta.ca

Michele Miscione
Dipartimento di Informatica e Automazione
Universitá Roma Tre
Rome, Italy
miscione@dia.uniroma3.it

ABSTRACT

Nowadays data is disseminated in a number of different sources, from databases systems to the Web, from a traditional structured organization (relational) to a semi-structured (XML), up to the unstructured ones (text in Web documents). Although availability of data is constantly increasing, one principal difficulty users have to face is to find and retrieve the information they are looking for. To this aim keywords search based systems are increasingly capturing the attention of researchers. In this paper, we present Yaanii[1], a tool for the effective Keyword Search over semantic datasets. It is based on a novel keyword search paradigm for graph-structured data, focusing in particular on the RDF data model. While many techniques search the best answer trees, we propose an effective algorithm for the exploration and computation of all matching subgraphs. We provide a clustering technique that identifies and groups graph substructures based on template match. A scoring function, IR inspired, evaluates the relevance of the substructures and the clusters. A strong point of our approach is that the ranking supports the generation of Top-k solutions during its execution.

1. INTRODUCTION

Keyword-based search approaches have the huge benefit that users can ignore both the language and the structure of the data they are going to query. A keyword based search engine returns a list of candidate pages, documents or set of data that match keywords provided in input. Then a user has to dedicate time and efforts navigating each result returned from the engine in order to discover the desired information, i.e. the answer he is looking for. Therefore, attention around searching and query processing of graph-structured data continue to increase as the Web, XML documents and even relational database can be represented as a graph. Current approaches rely on a combination of IR and tree/graph exploration techniques whose goal is to rank results according to a relevance criterion. Keyword search on tree-structured data counts a good number of approaches already [4, 5]. Actual efforts [3, 6] focus on RDF data querying, given the great momentum of *Semantic Web* in which Web pages carry information that can be read and understood by machines in a systematic way. Simplifying, a generic approach first identifies the parts of the data structure containing the keywords of interest, possibly by using an indexing system or a database engine, then explores the data structure in order to discover a connection between such identified parts. The common exploration paradigm is similar to the triple of RDF, that is $\langle subject, property, object \rangle$. Candidate solutions, built out of found connections, are then all generated and finally ranked through a scoring function. To return the top-k best solutions, pruning techniques reducing the list of candidate solutions down to those whose score is above a threshold are implemented. In this framework to achieve efficiency above all, current algorithms compute the best answers only in an approximate way. This is because they use an exploration paradigm that is inefficient and the scoring function takes place only when solutions were generated all together. Moreover pruning techniques can have a sensible impact in both the quality of the solutions, as low scoring results are not shown or even computed, as well as on efficiency, as an early pruning reduces the space of candidate solutions to investigate.

In [2] we proposed a novel approach to keyword search in the graph-structure data in a RDF representation. The main contributions of our approach are:

- A clustering technique that identifies and groups graph substructures based on *template* match. The idea is to group paths with respect to the *template* (i.e schema) they correspond to. A solution is a composition of paths belonging to different clusters. In this way we avoid the exploration of overlapping solutions and we build cleaner results for the users, gaining in terms of computation cost. Usually, the most promising algorithms of an efficient solution for keyword based search are in PTIME class complexity. To this aim, in [1] we demonstrated how Yaanii is more efficient with respect to the others, presenting a quadratic complexity as upper-bound.
- An algorithm that ranks solutions while it builds the solutions. Unlike most of the approaches to keyword search, that first identify all the solutions and then rank them according to a function, our approach leverages on the clusters to assembly a solution starting with the most relevant path in the most relevant cluster. As a result, the most relevant solution is the first to come out of the algorithm, then decreasing monotonically to the less relevant solutions. This allows users to explore the returned solutions, starting with the most relevant, while the elaboration of remaining solutions is undergoing.

2. AN ARCHITECTURE OF REFERENCE

We implemented our approach into a tool, called Yaanii. A flexible architecture of the system was design, as shown in Figure 1.

[1] Yaanii, literally "path" in Sanskrit.

Appears in the Proceedings of the 1st Italian Information Retrieval Workshop (IIR'10), January 27–28, 2010, Padova, Italy.
http://ims.dei.unipd.it/websites/iir10/index.html

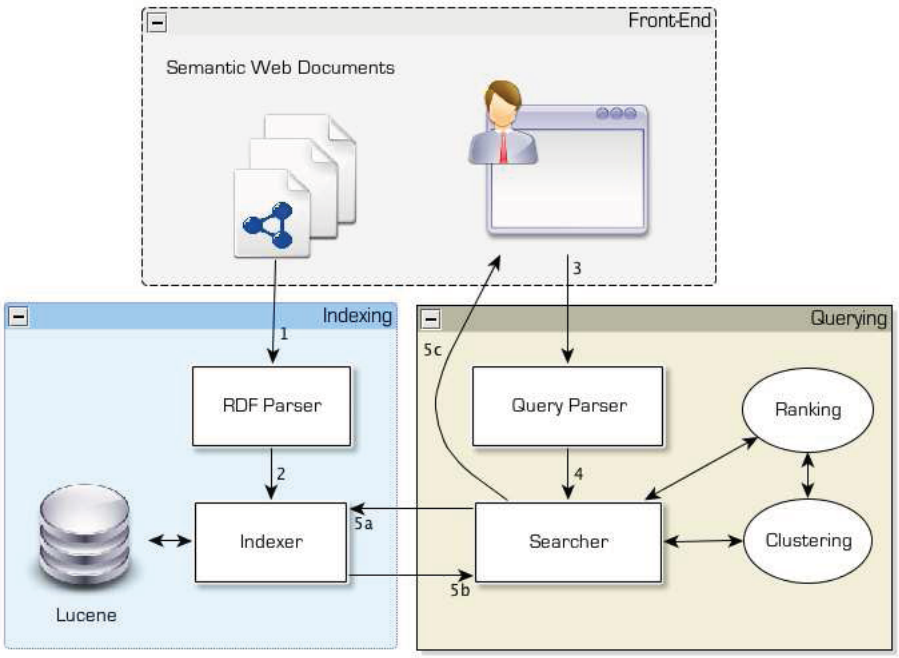

Figure 1: Architecture of Yaanii

It serves as a logical view of how the system looks like. This is a typical use scenario of the system:

1. The *RDF Parser* takes as input a collection of RDF Documents and parses them into triples. Here we use the Jena framework [2];

2. The *Indexer* builds an index on top of the triple collections to achieve structural information useful for the query process. Here the indexing is supported by Lucene[3] and WordNet [4]. The last allows query expansion;

3. A user performs a query through a *GUI helper*, handling events and the query itself;

4. The parsed query is given to the *Searcher* for processing;

5. The Searcher processes the query over the Indexed Resource Base and returns the search result to the caller. It communicates with the Indexer to extract the instances matching input keywords (i.e. informative paths), group them into clusters and compose elements from clusters into the final solutions (i.e. subgraph structures). Each structure (i.e. path, cluster and solution) is evaluated by a scoring function.

3. CONCLUSION AND FUTURE WORKS

We presented Yaanii, a tool for effective Keyword Search over semantic datasets. It is based on a clustering technique and a scoring function that support the generation of Top-k solutions during its execution in the first k steps.

From a theoretical point of view, future directions focus on improving the search algorithm of Yaanii to reach a linear time complexity. From a practical point of view, we would improve the indexing capabilities by embedding Lucene into a DBMS (e.g. Oracle) and provide a query-by-example interface to support the user to perform the query and navigate the results.

4. REFERENCES

[1] P. Cappellari, R. De Virgilio, M. Miscione. Keyword based Search over Semantic Data in Polynomial Time. In *Technical Report RT-DIA-160, Universitá Roma Tre, Rome, Italy*, 2009.

[2] R. De Virgilio, P. Cappellari, M. Miscione. Cluster-based exploration for Effective Keyword Search over Semantic Datasets. In *Proc. of the 28th International Conference on Conceptual Modeling (ER'09), Gramado, Brazil*, 2009.

[3] H. He, Wang, H., Yang, J., Yu, P.S. Blinks: ranked keyword searches on graphs. In *Int. Conf. on Management of Data (SIGMOD'07), China*, 2007.

[4] B. Kimelfeld, Sagiv, Y. Finding and approximating top-k answers in keyword proximity search. In *Int. Symposium on Principles of Database Systems (PODS'06), USA*, 2006.

[5] F. Liu, Yu, C.T., Meng, W., Chowdhury, A. Effective keyword search in relational databases. In *Int. Conf. on Management of Data (SIGMOD'06), USA*, 2006.

[6] T. Tran, Wang, H., Rudolph, S., Cimiano, P. Top-k exploration of query graph candidates for efficient keyword search on rdf. In *Int. Conf. on Data Engineering (ICDE'09), China*, 2009.

[2] http://jena.sourceforge.net/

[3] http://lucene.apache.org/

[4] http://wordnet.princeton.edu