1 Introduction

1.1.Motivation

Keyword-based queries over semi-structured data are an essential function of modern information management. When this data is available as Linked Data, it can be abstracted as a graph. Generally speaking, the results of queries over this type of data are sub-structures of the graph that contain the keywords searched in the nodes. The task of indexing and finding the individual nodes containing those keywords is relatively inexpensive and has well-established tools available for it.

Determining the connections between those selected nodes, however, is a complex problem that requires expensive and time-consuming graph explorations, and it must be solved on-the-fly at query processing time. And the scale at which those queries must be solved grows as more and more data is made available to the Web of Data. On top of that, to address the possibility of merging databases in the future, and to execute queries that encompass databases that had originally different schemas, those queries must rely only on the most basic format of the Linked Data model, the subject-predicate-object format.

1.2.Assumptions

This work is based on the proposal of de Virgilio [1] to a representation model of RDF graphs that uses an algebraic model based on tensorial calculus. By adopting this approach, one is capable of querying the RDF graph based solely on the data of the subject-predicate-object triples, without having to make assumptions about the schema or stored data patterns. It also addresses the issue of solving the queries on-the-fly by indexing the RDF graph in a sparse matrix that is built by pre-computing the time-consuming explorations needed. The limitation of de Virgilio's approach, however, is that it does not scale up to the dimensions of the Web of Data.

In this dissertation, we built upon the idea of using a tensor matrix to represent large RDF graphs, named tensor-based approach for shortness, as

follows. First, we propose a simplified version of how to build such matrix. De Virgilio's solution is again the basis for this step, but here we also make use of distributed resources to speed up its construction in contrast to his proposal. Then, we propose a distributed storage and retrieval solution to the indexes and the sparse matrix described in the tensor-based proposal. The consumption of the retrieved data and translating that to user-friendly interfaces falls outside the scope of this work. This dissertation focuses on leveraging the uses of such matrix in creating distributed datastores for RDF graphs to allow for the use of parallel computing techniques to handle the RDF graph more efficiently.

In this light, it is important to clarify that the present work makes the following assumptions:

- On-the-fly extractions of substructures of the RDF graph, as demonstrated by de Virgilio [1], are expensive to compute and do not necessarily retrieve useful data;
- Once extracted in the sparse matrix format, the RDF data can still be rebuilt and provide meaningful responses to queries as exemplified on de Virgilio's proposal.

Based on the above, we explore the tensor-based approach as a means to store RDF data in a distributed way. This organization allows for the storage of very large datasets and, more importantly, the use of parallel tools and techniques to deal with data more efficiently. In particular, we explore the use of the MapReduce technique to improve keyword based search. Better and faster results with keyword-based queries enable new insights into the data and facilitate access to the ever growing content of the Web of Data.

1.3.Contribution

The contributions of this work are twofold:

1. Provide a mechanism to store large RDF graphs in a distributed way. Using the tensor matrix representation as input, we developed a mechanism that divides (shards) the RDF graph and persist data in separate, distributed datastores.
2. Provide a scalable keyword-based search mechanism for RDF graphs. We use parallel processing techniques, notably the

MapReduce model, to provide a scalable solution to the task of building keyword indexes for RDF datasets.

1.4.Structure

This dissertation is structured as follows. On Chapter 2, we provide an overview of the topics that are fundamental to the understanding of this work. They include the use of RDF, Linked Data in particular, as an information structure and the use of Cloud Computing techniques. We provide an overview of the RDF model, discuss some of the fundamental ideas behind Linked Data, as well as the exponential growth of the Web of Data. The latter leads to a discussion on non-relational databases, their classification, and how they can fill the role of a scaled up storage system for large datasets in the Web. On a separate partition of Chapter 2, we discuss the evolving Cloud Computing model, with a focus in the enabling of the MapReduce model. We provide an overview of the technique emphasizing its deployment in Web environments, combined to the use of the available public cloud computational resources, e.g., Amazon Web Services (AWS) and Microsoft Azure.

On Chapter 3, we provide an overview of Linked Data keyword search. We discuss the problems and challenges of searching and extracting the correct amount of information from a large scale RDF graph. We then discuss and propose a simplified set of definitions of the tensor-based approach proposed by de Virgilio.

On Chapter 4, we analyse the steps and queries necessary to any system implementing the tensor matrix representation based approach. We extract the shared memory and functionalities required to build a scaled up system and, based on these, propose an architecture that matches these requirements.

On Chapter 5, we present and detail a possible implementation of the architecture proposed on Chapter 4.

On Chapter 6, we present our concluding remarks, discuss our contributions in the light of related works and point to future work.