Towards Scalingless Generation of Formal Contexts from an Ontology in a Triple Store

Frithjof Dau, SAP AG, Walldorf, Germany

ABSTRACT

The EU-funded research project CUBIST investigates how Formal Concept Analysis can be applied as a Visual Analytics tool on top of information stored in a Triple Store (TS). This paper provides first steps for utilizing SPARQL in order to generate formal contexts out of the data in the TS, where the emphasis is put on using object-properties between individuals. Thus it complements FcaBedrock, which will be used in CUBIST as well and focuses on the scaling of datatype-properties between individuals and literals. It is discussed how the approaches of this paper and FcaBedrock can be combined.

Keywords: Conceptual Scaling, Formal Concept Analysis, Semantic Technologies, Semantic Web, SPARQL Protocol and RDF Query Language (SPARQL), Triple Store (TS)

1. INTRODUCTION

CUBIST is an EU funded research project with an approach that leverages (Business Intelligence) BI to a new level of precise, meaningful and user-friendly analytics of data by following a best-of-breed approach that combines essential features of Semantic Technologies, Business Intelligence and Visual Analytics based on FCA (Formal Concept Analysis). CUBIST aims to:

• Provide novel ways of applying visual analytics based on meaningful diagrammatic representations.

The Visual Analytics part of CUBIST is complementing traditional BI-means by utilizing Formal Concept Analysis (FCA) for analyzing the data in the triple store. FCA is a well-known theory of data analysis which allows to conceptually clustering objects with respect to a given set of attributes and then visualize the (lattice-ordered) set of clusters, e.g. by means of Hasse-diagrams. The starting point of FCA is a formal context \((O,A,I)\) consisting of a set \(O\) of formal objects, a set \(A\) of formal attributes, and an incidence-relation

DOI: 10.4018/ijcssa.2013010102
$M \subseteq O \times A$ between the formal objects and attributes. There exists a variety of tools to carry out analysis of formal contexts (e.g. conexp, Lattice Miner, or conflexplore), but nearly all of them take a formal context as input. Real data to be analyzed, however, often comes in different forms:

- Conceptually, often attributes are not binary, but have values like numbers, strings, or dates (e.g. We have many-valued attributes);
- Technically, data can come in form of csv-files, databases, triple stores, etc.

For dealing with many-valued attributes, the best-known and most-used method is conceptual scaling (Ganter & Wille, 1989). Essentially, for a given-many valued attribute, a conceptual scale is a specific context with the values of the many-valued attribute as formal objects. The choice of the formal attributes of the scale is a question of the design of the scale: The formal attributes are meaningful attributes to describe the values; they might be different entities or they might even be the values of the property again. Using a conceptual scale, a dataset with a many-valued attribute can be “translated” into a formal context, where the objects are the objects of the dataset and the attributes are the attributes of the conceptual scale, and the derived formal context can be analyzed with FCA.

From the technical point of view, there are (to the author’s knowledge) essentially two tools which allow for scaling real datasets:

- Toscanaj (Becker & Correia, 2005) is a suite of tools which allows to creating conceptual scales out of data from a relational database and then interactively visualizing and exploring the generated concept lattices;
- FcaBedrock (Andrews, 2009; Andrews & Orphanides, 2010) is a tool which converts csv-files into formal contexts. It is “taking each many-valued attribute and converting it into as many Boolean attributes as it has values and converting continuous values using ranges.” (Andrews & Orphanides, 2010).

The approach of Toscanaj is a two-step approach: First, in the design phase of a conceptual information system, the conceptual scales are created by an FCA-experienced designer. In the run-time phase, these scales are used by the user to explore the data. The downside of this approach is that the scales are predefined in the design-phase, which in turn implies that the lattice-structure of the scales is fixed, thus Toscanaj does not really allow new structural insights into the data to be obtained. In the beginning of CUBIST, a modified version of Toscanaj, called ToscanajTS (“TS” for “Triple Store”) has been developed which acts on a triple store instead of a database (Dau & Sertkaya, 2011a; Dau & Sertkaya, 2011b). ToscanajTS shows the applicability of FCA on top of a triple store, but it inherits the above discussed downsides of Toscanaj as well.

The approach of FcaBedrock slightly differs from Toscanaj. Similar to Toscanaj, there is information needed on how to convert real data into a formal context. As stated by Andrews and Orphanides, 2010: “FcaBedrock solves these problems by documenting data conversions in re-usable, editable, meta-data-files called bedrock files.” The difference to Toscanaj is that the formal attributes of the generated formal context are not manually defined during the design phase, but they are created on the fly from the real data and the meta-information in the bedrock files. In this respect, FcaBedrock better serves the purpose to generate formal contexts out of the real data on the fly. For CUBIST, though, there are the following disadvantages: First, FcaBedrock puts a focus on many-valued attributes, but—as it will be discussed in the next sections—there are indeed possibilities to obtain binary relations directly (without scaling) out of the information in a triple store. Secondly, the existing version of FcaBedrock needs csv-files...
Related Content

Knowledge Acquisition Modeling Through Dialogue Between Cognitive Agents
[www.igi-global.com/article/knowledge-acquisition-modeling-through-dialogue/2414?camid=4v1a](www.igi-global.com/article/knowledge-acquisition-modeling-through-dialogue/2414?camid=4v1a)

Induction as a Search Procedure
[www.igi-global.com/chapter/induction-search-procedure/5323?camid=4v1a](www.igi-global.com/chapter/induction-search-procedure/5323?camid=4v1a)
Self Adaptive Particle Swarm Optimization for Efficient Virtual Machine Provisioning in Cloud
[www.igi-global.com/article/self-adaptive-particle-swarm-optimization/54065?camid=4v1a](www.igi-global.com/article/self-adaptive-particle-swarm-optimization/54065?camid=4v1a)

Knowledge Extraction Through Page Rank Using Web-Mining Techniques for E-Business: A Review
[www.igi-global.com/chapter/knowledge-extraction-through-page-rank-using-web-mining-techniques-for-e-business/178294?camid=4v1a](www.igi-global.com/chapter/knowledge-extraction-through-page-rank-using-web-mining-techniques-for-e-business/178294?camid=4v1a)