Ontology Merging with Formal Concept Analysis

Gerd Stumme

Chair of Knowledge and Data Engineering, University of Kassel
Wilhelmshöher Allee 73, D–34121 Kassel, Germany
http://www.kde.cs.uni-kasel.de; stumme@cs.uni-kassel.de

Abstract. Ontologies have been established for knowledge sharing and are widely used as a means for conceptually structuring domains of interest. With the growing usage of ontologies, the problem of overlapping knowledge in a common domain becomes critical. In this short paper, we address two methods for merging ontologies based on Formal Concept Analysis: FCA–MERGE and ONTEX.

FCA–MERGE is a method for merging ontologies following a bottom-up approach which offers a structural description of the merging process. The method is guided by application-specific instances of the given source ontologies. We apply techniques from natural language processing and formal concept analysis to derive a lattice of concepts as a structural result of FCA–MERGE. The generated result is then explored and transformed into the merged ontology with human interaction.

ONTEX is a method for systematically structuring the top-down level of ontologies. It is based on an interactive, top–down knowledge acquisition process, which assures that the knowledge engineer considers all possible cases while avoiding redundant acquisition. The method is suited especially for creating/merging the top part(s) of the ontologies, where high accuracy is required, and for supporting the merging of two (or more) ontologies on that level.

1 Introduction

Ontologies have been established for knowledge sharing and are widely used as a means for conceptually structuring domains of interest. With the growing usage of ontologies, the problem of overlapping knowledge in a common domain occurs more often and becomes critical. Domain-specific ontologies are modeled by multiple authors in multiple settings. These ontologies lay the foundation for building new domain-specific ontologies in similar domains by assembling and extending multiple ontologies from repositories.

The process of *ontology merging* takes as input two (or more) source ontologies and returns a merged ontology based on the given source ontologies. Manual ontology merging using conventional editing tools without support is difficult, labor intensive and error prone. Therefore, several systems and frameworks for supporting the knowledge engineer in the ontology merging task have recently been proposed [Ho98,Ch00,NM00,MFRW00]. The approaches rely on syntactic

Dagstuhl Seminar Proceedings 04391
Semantic Interoperability and Integration
http://drops.dagstuhl.de/opus/volltexte/2005/49

and semantic matching heuristics which are derived from the behavior of ontology engineers when confronted with the task of merging ontologies, i. e., human behavior is simulated. Although some of them locally use different kinds of logics for comparisons, these approaches do not offer a structural description of the global merging process.

We developed two approaches for merging and/or creating ontologies based on Formal Concept Analysis (FCA) [Wi82]: FCA-Merge (joint work with A. Mädche) and OntEx (joint work with B. Ganter). FCA provides us with a structural description of the merging process. It is a mathematical formalization of the concept of 'concept'. In FCA–Merge, FCA is used as a technique for conceptual clustering. In OntEx, it is used as a technique for interactive knowledge acquisition. An introduction to FCA can be found at [GW99].

The article summarizes work presented at IJCAI 2001 [SM01] and ICCS 2003 [GS03]. In the next section, FCA–Merge is presented. In Section 3, OntEx is presented.

2 FCA–Merge: Bottom-Up Merging of Ontologies

FCA–Merge is a method for merging ontologies following a bottom-up approach and offering a global structural description of the merging process. For the source ontologies, it extracts instances from a given set of domain-specific text documents by applying natural language processing techniques. Based on the extracted instances we derive a concept lattice. The concept lattice provides a conceptual clustering of the concepts of the source ontologies. It is explored and interactively transformed to the merged ontology by the ontology engineer. The approach is described in detail in [SM01].

FCA–Merge is based on application-specific instances of the input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ that are to be merged. The overall process of merging two ontologies is depicted in Figure 1 and consists of three steps, namely *(i)* instance extraction and the computation of two formal contexts $\mathbb{K}_1$ and $\mathbb{K}_2$, *(ii)* the FCA-Merge core algorithm that derives a common context and computes a concept lattice, and *(iii)* the generation of the final merged ontology based on the concept lattice.

The method takes as input the two ontologies and a set D of natural language documents. The documents have to be relevant to both ontologies, so that the documents are described by the concepts contained in the ontology. The documents may be taken from the target application which requires the final merged ontology. From the documents in D, we *extract instances.* This automatic knowledge acquisition step returns, for each ontology, a formal context indicating which ontology concepts appear in which documents.

The extraction of the instances from documents is necessary because there are usually no instances which are already classified by both ontologies. However, if this situation is given, one can skip the first step and use the classification of the instances directly as input for the two formal contexts.

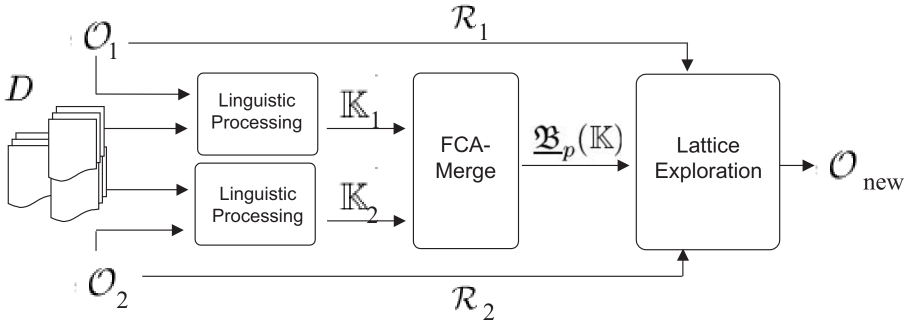

Fig. 1. Ontology Merging Method

The second step of the approach comprises the FCA–MERGE core algorithm. The core algorithm merges the two contexts and computes a concept lattice from the merged context using the TITANIC algorithm [STBPL02]. More precisely, it computes a *pruned concept lattice* which has the same degree of detail as the two source ontologies.

Instance extraction and the FCA–MERGE core algorithm are fully automatic. The final step of *deriving the merged ontology* from the concept lattice requires human interaction. Based on the pruned concept lattice and the sets of relation names $\mathcal{R}_1$ and $\mathcal{R}_2$, the ontology engineer creates the concepts and relations of the target ontology. The ontology engineering environment OntoEdit provides graphical means for supporting this process.

Certainly, high quality results of the merging process will always need some human involved, who is able to make judgments based on background knowledge, social conventions, and purposes. Thus, all merging approaches aim at supporting the knowledge engineer, and not at replacing him. Our approach differs from the related work stated above in that it provides, for one part of the merging process, an algorithm with a well-defined description of the output in terms of the input. If the knowledge engineer commits to this description, he is guaranteed to obtain the expected results. FCA–MERGE may of course also be included in a heuristics-based approach as a reliable building block.

3 OntEx: Creation and Merging of Ontology Top-Levels

ONTEX (Ontology Exploration) is a method for supporting the two tasks of creating and merging ontologies [GS03]. It relies on the knowledge acquisition technique of Attribute Exploration [GW99] as developed in the mathematical framework of Formal Concept Analysis. ONTEX guarantees that the knowledge engineer considers all relevant combinations of concepts for the creation/merging process but avoids redundant acquisition.

The first task we address is the *creation of a new ontology*. When ontologies have to be created from scratch, the user needs some guidance how to start. Especially the very first decisions have strong impact on the result, as they determine the overall structure of the ontology. Creation of ontologies from scratch is usually performed top–down. First the most general concepts of the ontology are selected. More specific concepts are then added by classifying them in the already present structure. OntEx supports one part of this creation process, namely the structuring of the conceptual hierarchy on the top–level concepts, and the creation of new concepts on the next level by providing suggestions based on an interactive exploration of the existing structure. The second task we address is *ontology merging*. OntEx uses the same interactive knowledge acquisition technique also for merging of the top–level concepts, where design decisions have the most impact on the overall structure of the target ontology.

The OntEx approach can be split into three steps:

1. initialization of the exploration contexts,
2. the exploration process,
3. further processing.

In the initialization step, first the user has to provide an initial set of concepts she considers to be relevant. For the merging task, these are just the concepts of the start ontologies. How to obtain this initial set for the creation task is beyond the scope of this paper. It may be determined by other knowledge acquisition techniques as for instance described in [SSSS01]. Next one formalizes both the background knowledge – especially the known subsumptions – (and eventually some counter–examples as described below which are known apriori).

The exploration process comprises the exploration dialogue with the user, consisting of questions as described below. At the end of this process, the lattice of all conjunctions of the input concepts is determined. It contains all input concepts and some new concepts constructed during the process.

The exploration process is based on the knowledge acquisition technique called *Attribute Exploration* [GW99]. For a given set of ontology concepts, it determines the lattice of all conjunctions of these concepts. In an interactive process, it asks the user questions of the kind "Is the conjunction of the concepts c_1, c_2, ..., and c_n a subconcept of all of the concepts c'_1, c'_2, ..., and c'_m?" with $n, m \geq 0$. The user can either accept or reject the subsumption. If she rejects, she has to provide a counter-example, i. e., a new object [or a new concept] which belongs to the extent of [is a subconcept of, resp.] all concepts c_1, c_2, ..., and c_n but not to at least one of the concepts c'_1, c'_2, ..., and c'_m. In this way, the list of subsumptions as well as the list of counter-examples grows iteratively, until all pairs of concepts are either in a subconcept–superconcept–relation or there is a counter-example prohibiting this.

In the third phase, the user can modify the resulting hierarchy using any ontology editor, eventually supported by some heuristic approach.

The use of OntEx guarantees that the knowledge engineer considers all relevant possibilities both for the creation and for the merging task. However, this guarantee is paid with a certain workload for the knowledge engineer, making it applicable only to relatively small parts of the ontologies at hand. Therefore we propose a two–step, top–down approach. The first step aims at reliably creating/merging the top–part(s) of the ontologies with high accuracy, using ONTEX. In the second step, any heuristics–based approach can be used for creating the remainder of the target ontology with less user interaction. This two–step approach allows for high accuracy for the design of the top–level ontology, which has large impact on the global structure of the resulting ontology, as it is more difficult to modify in a later phase than local decisions on a lower level of detail. On the other hand, it restricts the comparatively high workload on the user to the first, critical phase.

References

[Ch00] H. Chalupsky: OntoMorph: A translation system for symbolic knowledge. *Proc. 7th Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR'2000)*, Breckenridge, Colorado, USA, April 2000,471–482

[GS03] B. Ganter, G. Stumme: Creation and Merging of Ontology Top-Levels. In: A. de Moor, W. Lex, B. Ganter (eds.): *Conceptual Structures for Knowledge Creation and Communication.* Proc. ICCS '03. LNAI 2746, Springer, Heidelberg 2003, 131–145

[GW99] B. Ganter, R. Wille: *Formal Concept Analysis: Mathematical Foundations.* Springer, Heidelberg 1999. (Translation of: *Formale Begriffsanalyse: Mathematische Grundlagen.* Springer, Heidelberg 1996.)

[Ho98] E. Hovy: Combining and standardizing large-scale, practical ontologies for machine translation and other uses. *Proc. 1st Intl. Conf. on Language Resources and Evaluation*, Granada, Spain, May 1998

[MFRW00] D. L. McGuinness, R. Fikes, J. Rice, and S. Wilder: An environment for merging and testing large Ontologies. *Proc. 7th Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR'2000)*, Breckenridge, Colorado, USA, April 2000, 483–493

[NM00] N. Fridman Noy, M. A. Musen: PROMPT: algorithm and tool for automated ontology merging and alignment. *Proc. 17th Natl. Conf. on Artificial Intelligence (AAAI'2000)*, Austin, TX, July/August 2000, 450–455

[SSSS01] S. Staab, H.–P. Schnurr, R. Studer, Y. Sure: Knowledge Processes and Ontologies. *IEEE Intelligent Systems* **16**(1), 2001

[SM01] G. Stumme, A. Mädche: FCA–Merge: Bottom-Up Merging of Ontologies. *Proc. 17th Intl. Conf. on Artificial Intelligence (IJCAI '01).* Seattle, WA, USA, 2001, 225–230

[STBPL02] G. Stumme, R. Taouil, Y. Bastide, N. Pasqier, L. Lakhal: Computing Iceberg Concept Lattices with Titanic. *J. on Knowledge and Data Engineering* **42**(2), 2002, 189–222

[Wi82] R. Wille: Restructuring lattice theory: an approach based on hierarchies of concepts. In: I. Rival (ed.): *Ordered sets.* Reidel, Dordrecht–Boston, 445–470