# Semi-automatic Ontology Engineering using Patterns

Eva Blomqvist

Jönköping University, Jönköping, Sweden
blev@jth.hj.se

**Abstract.** Many enterprise systems face the same kind of information processing problems that exist on the web in general, and creating semantic solutions often involve constructing an enterprise ontology. Ontology engineering in turn needs to be semi-automatic in order to reduce the effort and need for expertise. By introducing knowledge reuse in ontology construction the construction effort can be further reduced and the quality of the ontology output increased. The proposed research focuses on a hybrid approach for ontology construction based on the methodology of case-based reasoning in combination with ontology patterns.

## 1 Introduction

Semantic Web techniques are not only applicable to the "public" web, similar problems can be found also in the business world, for example on large company intranets. Development of semantic applications for enterprises most often involve constructing an enterprise ontology for the company in question. Manual ontology engineering is a tedious and complex task, therefore most recent research focus on semi-automatic approaches. One issue addressed by our research is automation throughout the construction process. Another issue is knowledge reuse; common practises of the business world should be exploited, as well as best practises in ontology engineering. To learn from past experience partial solutions from already constructed ontologies can be collected. Our intention is to develop a hybrid approach (named OntoCase) that combines the learning viewpoint from case-based reasoning (CBR) with more consensual knowledge reuse through patterns.

## 2 Background and Related Work

Our research is not restricted to a specific ontology representation formalism but assumes for some parts the possibility to represent the ontology as a (semantic net-like) graph. The research focuses on application ontologies within enterprises, here denoted enterprise application ontologies, used mainly for structuring and retrieval of information.

### 2.1 Ontology Engineering

Recent developments in ontology engineering involve semi-automatic ontology construction, or ontology learning (OL). Most OL approaches focus on techniques for text analysis in order to extract mainly concepts and relations from a text corpus. Existing approaches are for example Text2Onto [1], OntoGen [2] and Abraxas [3]. One major issue is the problem of background information not being explicitly stated in a text document. Some approaches use additional sources like WordNet (see [4]), or the web, to find and validate missing information. Patterns can also serve as such a source of additional information and structure. Recent research also tries to apply OL techniques throughout the complete ontology life-cycle and treat sets of interconnected ontologies (as suggested by [5]).

Four abstraction levels of patterns describing the internal structure of an ontology have been discussed in [6] (syntactic, semantic, design and architecture patterns). Here we focus on design and architecture patterns for semi-automatic use. An ontology design pattern is an ontology template intended to construct a part of some ontology, which is self-contained and comprised of a set of ontology primitives. Related work in ontology design patterns focus on templates mainly for manual use (e.g. [7]), but recent research efforts (like [5]) intends to also extend this into OL. The architecture of an ontology describes the components from which it is comprised, their dependencies and topology. In our method, architecture patterns will mainly be sets of constraints guiding and restricting the composition of the ontology from design patterns.

One major task involves how to assess pattern relevance. This problem is quite similar to ranking in an ontology search engine, as for example in [8], and also uses basic techniques from ontology matching (see the survey in [9]). The main differences are that in OntoCase we have a much richer input structure (the ”query”) as compared to ontology search, whereas compared to the general case of ontology matching we have some very specific characteristics of the input ontologies and additionally specific requirements on the matching result.

### 2.2 Case-based Reasoning

Case-based reasoning (or CBR, see [10]) is a methodology for using previous experience to solve new problems. A case is a problem situation, and previously encountered cases are stored in a case base. The CBR process is viewed as a cycle of four phases. The first phase is retrieval, where a new case arrives, its representation is derived and used to find relevant stored cases. The second phase is reuse, where the retrieved case(s) are adapted and used to form a new solution. Next, the solution is evaluated and revised, and finally relevant parts of the new solution are retained and stored. Our approach falls within textual CBR (TCBR), where parts of cases or solutions are natural language texts (see [11]). The first and second phase of OntoCase corresponds to current research questions of TCBR as stated in [11]. The CBR idea is quite similar to the idea of using patterns, but CBR takes an even more pragmatic view by basing the solutions only on previous experience. We would like to construct a hybrid approach that exploits both best practises and new experiences.

## 3 Research Questions

In our research we have set a number of long-term goals based on the open issues in current OL research. From these long-term goals general research questions have been derived:

- How can the effort of constructing an enterprise application ontology be reduced?
  - How can CBR improve semi-automatic ontology construction?
- How can knowledge and experience be reused in ontology engineering?
  - How can ontology patterns be exploited in the semi-automatic ontology construction process?

Based on the research questions above the following hypotheses have been stated:

- Automation reduces the total construction effort.
  - CBR gives a framework for further automation of ontology construction, compared to related semi-automatic approaches that exist today.
- Domain knowledge and engineering experience can be reused through ontology patterns.
  - The CBR methodology together with patterns can improve the quality of generated ontologies, compared to existing semi-automatic approaches.

To test the hypotheses OntoCase is being developed based on CBR and the notion of ontology patterns. To verify the hypotheses the method must be evaluated and compared to related OL approaches. So far only minor parts of the approach have been evaluated. For the evaluation of the overall approach we envision the use of a "gold standard" ontology developed within one of our projects in cooperation with industry, to compare our results with achievable results of existing available OL systems.

## 4 Proposed Approach

The basis of a CBR approach is the case base, which in OntoCase corresponds to a pattern catalogue (pattern base), containing both ontology design and architecture patterns. The design patterns are represented as small ontologies (see example in [12]). Partly due to the lack of available enterprise ontologies the patterns have so far been constructed based on other sources, like data model patterns and textbooks on organisations. The architecture patterns are sets of constraints on the combination of design patterns. Although the CBR cycle can be illustrated as a linear process, iteration is of course present internally in the phases, also it is intended that the phases can be applied individually depending on already available input.

The retrieval phase constitutes of analysing the input text corpus and deriving its representation, then matching this to the pattern base and selecting appropriate patterns. The reuse phase concerns the adaptation of the patterns, combining them into an ontology. The revise phase includes extending the ontology, based on evaluation results. Retaining patterns includes the discovery of new patterns and improving existing patterns. So far retrieval and reuse has been the main focus of research, revise and retain are still future work.

### 4.1 Achieved Results

In [12] an industry project was used as an application case for a simple implementation of the first two phases. The evaluations described in [13] gave rise to the current research questions and hypotheses as stated previously. The main problems that were discovered could be connected to missing information in the text corpus and pattern base, and the lack of a general architecture. We envision the refinement of the approach as described below, and the additional two phases (revise and retain) to remedy many of those problems.

The initial step of the retrieval phase, extracting a representation of the text corpus, is considered mainly outside the scope of this work. We feel this is an area where already a lot of approaches exist. A small initial evaluation was conducted, comparing recent OL systems with more basic text analysis techniques available in standard components. The term output was compared to a manually constructed "gold standard". The OL system being tested almost doubled the precision compared to a combination of basic techniques, and improved recall around 30%. So far this is our motivation for trying to build on recent experimental algorithms instead of using only standard text analysis components.

The second step of the retrieval phase concerns choosing suitable patterns. For this, a pattern ranking scheme has been developed, partly inspired by ontology ranking as well as techniques in ontology matching. Compared to ontology search engines, our pattern ranking exploits a much richer input structure since the complete input representation is used. This has lead to a ranking scheme of four measures; concept coverage, relation coverage, density, and proximity.

The second phase (reuse) involves first composing the initial ontology from the patterns, essentially ontology merging. This has so far been a simple process applying heuristics for resolving overlap between patterns. Furthermore, the phase contains an enrichment step where the initial ontology is extended using parts of the input representation until a suitable coverage (over the input) has been reached. External knowledge sources, like WordNet, are used for placing the input primitives in the context of the resulting ontology without loss of quality.

In our approach there is an aspect of uncertainty inherent in all the described steps. For example, each ontology primitive of the input representation has a certain degree of confidence associated. The levels of confidence are transferred onto the constructed ontology. If a standard ontology representation is required, thresholds can be set or the ontology can be validated by a user.

### 4.2 Next Steps

The two last phases, revise and retain, are still not elaborated in our approach. The revision phase is intended to contain steps for evaluating and further revising the ontology, to increase the input coverage but also for example to reduce the level of redundancy and resolve possible conflicts in the ontology. The final phase of retaining patterns is concerned with collecting feedback for applied patterns, but also to discover new pattern candidates directly from the resulting ontology. Since this is denoted a hybrid approach, in addition to the pattern discovery step, patterns can be constructed manually (but the development of a manual pattern construction method is considered outside the scope of this work). The idea is to benefit both from the CBR view (reuse of solutions) and the pattern view (reuse of best practises).

## 5 Expected Contribution and Future Work

The main contributions of our ontology engineering approach are envisioned as both further automation of the ontology construction process, but in addition an increased quality of the produced ontology, as compared to existing OL approaches. This increased quality is mainly due to the use of patterns, representing both expert knowledge and previous experience, and the introduction of several revision steps during the four phases of the method.

Future work contains refinement of the first two phases, as well as implementation of the improvements in the prototype system. Also to treat the third and fourth phases in detail and include these in an implementation. Some open issues are how to automatically find and extract patterns, as well as how to extend the ontology in the revision phase. Probably external sources, like the web, have to be used to attach and validate missing pieces of the ontology. A major step in all research is of course evaluation of the approach, in this case to compare the result to the result of manual approaches as well as the related OL approaches stated earlier. These evaluations are so far not planned in detail.

## References


1. Cimiano, P.: Ontology Learning and Population from Text: Algorithms, Evaluation and Applications. Springer Science (2006)
2. Fortuna, B., Grobelnik, M., Mladenic, D.: Semi-automatic Data-driven Ontology Construction System. In: Proc. of IS-2006, Ljubljana, Slovenia (2006)
3. Iria, J., Brewster, C., Ciravegna, F., Wilks, Y.: An Incremental Tri-partite Approach to Ontology Learning. In: Proc. of LREC2006, Genoa (2006)
4. C. Fellbaum et al.: WordNet - An Electronic Lexical Database. MIT Press (1998)
5. NeON Website. Available at: http://www.neon-project.org/.
6. Blomqvist, E., Sandkuhl, K.: Patterns in Ontology Engineering: Classification of Ontology Patterns. In: Proc. of ICEIS2005, Miami Beach, Florida (2005)
7. Gangemi, A.: Ontology Design Patterns for Semantic Web Content. In: Proceedings of ISWC 2005. Volume 3729 of LNCS., Springer (2005) 262–276
8. Alani, H., Brewster, C.: Ontology Ranking based on the Analysis of Concept Structures. In: Proceedings of K-CAP'05, Banff, Alberta, Canada (2005)
9. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer Berlin Heidelberg (2007)
10. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues , methodological variations, and system approaches. AICom, IOS Press **7** (1994) 39–59
11. Weber, R.O., Ashley, K.D., Brüninghaus, S.: Textual case-based reasoning. The Knowledge Engineering Review **20**(3) (2006) 255–260
12. Blomqvist, E.: Fully automatic construction of enterprise ontologies using design patterns: Initial method and first experiences. In: Proc. of ODBASE. (2005)
13. Blomqvist, E., Öhgren, A., Sandkuhl, K.: Ontology Construction in an Enterprise Context: Comparing and Evaluating two Approaches. In: Proc. of ICEIS. (2006)