Learning OWL Class Expressions

Jens Lehmann

(Thesis Summary)

With the advent of the Semantic Web and Semantic Technologies, ontologies have become one of the most prominent paradigms for knowledge representation and reasoning. The currently most popular ontology language OWL, based on description logics, became a W3C recommendation in 2004 and a standard for modelling ontologies on the Web. In the meantime, many studies and applications using OWL have been reported in research, many of which go beyond Internet usage and employ the power of ontological modelling in other fields such as biology, medicine, software engineering, knowledge management, and cognitive systems.

However, recent progress in the field faces a lack of well-structured ontologies with large amounts of instance data due to the fact that engineering such ontologies requires a considerable investment of resources. Nowadays, knowledge bases often provide large volumes of data without sophisticated schemata. Methods for automated schema acquisition and maintenance are, therefore, sought. Furthermore, many classification and knowledge acquisition problems, e.g. the detection of chemical compounds causing cancer, can be handled by using the same techniques.

In order to leverage machine-learning approaches for solving these tasks, it is required to develop methods and tools for learning concepts in description logics or, equivalently, class expressions in OWL. In this thesis, it is shown that methods from Inductive Logic Programming (ILP) are applicable to learning in description logic knowledge bases. The results provide foundations for the acquisition of OWL ontologies, in particular in cases when extensional information (facts, instance data) is easily available, while corresponding intensional information (schema) is missing or not expressive enough to allow powerful reasoning over the ontology in a useful way. Such situations often occur when extracting knowledge from different sources, e.g. databases and wikis, or in collaborative knowledge engineering scenarios. It can be argued that being able to learn OWL class expressions is a step towards enriching OWL knowledge bases in order to enable powerful reasoning, consistency checking, and improved querying possibilities. In particular, plugins for OWL ontology editors based on learning methods are developed and evaluated in this work.

The developed algorithms are, of course, not restricted to ontology engineering and can handle other learning problems. Indeed, they lend themselves to generic use in machine learning in the same way as ILP systems do. The main difference, however, is the employed knowledge representation paradigm: ILP traditionally uses logic programs for knowledge representation, whereas this work rests on DLs/OWL. This distinction is crucial when considering Semantic Web applications as target use cases, as such applications hinge centrally on the chosen knowledge representation format for knowledge interchange and integration. The work in this thesis can be understood as a broadening of the scope of research and applications of ILP methods. This goal is particularly important since the number of OWL-based systems can be expected to increase rapidly in the near future.

The thesis starts by establishing the necessary theoretical basis and continues with the specification of algorithms. It also contains their evaluation and, finally, presents a number of application scenarios. The research contributions of this work are threefold:

The first contribution is a full analysis of desirable properties of refinement operators in description logics. Refinement operators are used to traverse the target search space and are, therefore, a crucial element in many learning algorithms. Their properties (completeness, weak completeness, properness, redundancy, infinity, minimality) indicate whether a refinement operator is suitable for being employed in a learning algorithm. The key research question is which of those properties can be combined. It is shown that there is no ideal, i.e. complete, proper, and finite, refinement operator for expressive description logics, which indicates that learning in description logics is a challenging machine learning task. A number of other new results for different property combinations are also shown. The need for these investigations has already been expressed in several articles prior to this PhD work. The theoretical limitations, which were shown as a result of these investigations, provide clear criteria for the design of refinement operators. In the analysis, as few assumptions as possible were made regarding the used description language.

The second contribution is the development of two refinement operators. The first operator supports a wide range of concept constructors and it is shown that it is complete and can be extended to a proper operator. It is the most expressive operator designed for a description language so far. The second operator uses the light-weight language $\mathcal{EL}$ and is weakly complete, proper, and finite. It is straightforward to extend it to an ideal operator, if required. It is the first published ideal refinement operator in description logics. While the two operators differ a lot in their technical details, they both use background knowledge efficiently.

The third contribution are the actual learning algorithms using the introduced operators. New redundancy elimination and infinity-handling techniques are introduced in these algorithms. According to the evaluation, the algorithms produce very readable solutions, while their accuracy is competitive with the state-of-the-art in machine learning. Several optimisations for achieving scalability of the introduced algorithms are described, including a knowledge base fragment selection approach, a dedicated reasoning procedure, and a stochastic coverage computation approach.

The research contributions are evaluated on benchmarks problems and in use cases. Standard statistical measurements such as cross validation and significance tests show that the approaches are competitive. Furthermore, the ontology engineering use case study provides evidence that the described algorithms can solve the target problems in practice.