Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Combining Ontology Class Expression Generation with Mathematical Modeling for Ontology Learning

Jedrzej Potoniec and **Agnieszka Ławrynowicz**
Institute of Computing Science, Poznan University of Technology
ul. Piotrowo 2, 61-381 Poznan, Poland
e-mail: {jpotoniec,alawrynowicz}@cs.put.poznan.pl phone: +48-61-665-30-26

Abstract

We present an idea of using mathematicall modelling to guide a process of mining a set of patterns in an RDF graph and further exploiting these patterns to build expressive OWL class hierarchies.

Introduction

Due to high cost of manual creation of ontologies, there have been many methods proposed for (semi-)automatic ontology learning (Lehmann and Völker 2014). Those methods employ various structured and unstructured sources. Recently, there is an interest in ontology learning from so called Linked Data (Bizer, Heath, and Berners-Lee 2009), where several approaches have been proposed, generally falling into one of two categories: logical top-down methods (e.g. (Fanizzi, D'Amato, and Esposito 2008)) and statistical bottom-up methods (e.g. (Völker and Niepert 2011)). The methods from the first category work in a supervised manner and thus they require positive and negative examples, that are hard to obtain in the open Web context of Linked Data. The proposed statistical methods are unsupervised, but due to their bottom-up nature, they are hard to focus on specific parts of the ontology and generate a lot of results that need to be further analyzed by experts.

This paper tackles a task in OWL ontology learning from Linked Data: the induction of a taxonomy of expressive class descriptions i.e. a branch of an expressive ontology rooted at a given class. On the one hand, we would like to apply a refinement operator to systematically grow class descriptions starting from a root class to focus the method on a particular ontology branch. On the other hand, the goal is also to have a method working in unsupervised manner that does not require training examples. In order to meet all of these requirements we propose a novel approach to ontology learning: a combination of a refinement operator based class induction methods with mathematical modeling. We create a mathematical model for the proposed ontology learning task and devise a method that interleaves two steps: i) an application of a refinement operator to generate a set of candidate classes, and ii) an application of our mathematical model to the set of generated classes to choose the optimal set of classes for further refinement.

Related work

In (Völker and Niepert 2011), a statistical method for schema induction is introduced, based on association rule mining. The method works in a bottom-up manner, and is thus hard to focus on specific parts of the ontology.

Early logical approaches are based on Inductive Logic Programming that combines machine learning and logic programming techniques in order to learn logical theories from examples and background knowledge. A number of proposed approaches to solve this task, are based on refinement operators, e.g. DL-FOIL (Fanizzi, D'Amato, and Esposito 2008) or algorithms implemented in DL-Learner (Lehmann 2009). Refinement operator, a function that computes pattern specializations/generalizations, allows for specifying constraints such as a class to start from. The drawback of such methods is that they require positive and negative examples, where the latter ones are often not available.

Potentially, an unsupervised frequent class description mining method could be used for ontology learning such as Fr-ONT (Ławrynowicz and Potoniec 2011). A general problem with using such method is that it may easily lead to many computed classes, since the algorithm first generates all valid class refinements and then also their permutations.

Preliminaries

To obtain set of refined patterns, we use the Fr-ONT-Qu algorithm, which purpose is to discover patterns in an RDF graph. Every pattern is a SPARQL query[1] with a single variable in its head (`?x` in examples below) and every other variable is connected to this variable by a chain of properties. Fr-ONT-Qu consists of a refinement operator and a strategy to select the best patterns for further refinement. Below a short example to cover Fr-ONT-Qu's most important aspects is presented.

An input of the algorithm is a declarative bias to limit a search space (i.e. classes and properties to use) and maximal number of iterations. Consider the declarative bias containing classes `PassengerTrain`, `CargoTrain` and property `hasEngine`.

[1] http://www.w3.org/TR/sparql11-query/

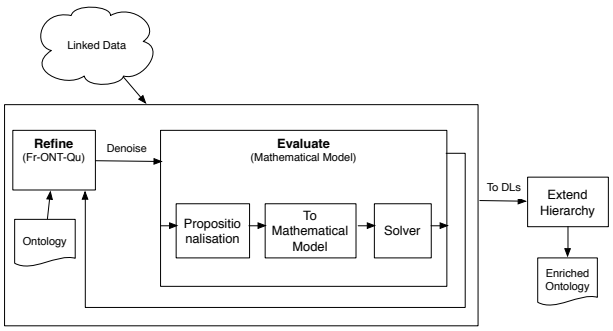

Figure 1: Overview of the proposed approach

1. Refine every pattern from the previous iteration by adding a single restriction for a variable already existing in the pattern. E.g. consider a pattern {`?x a :Train.`}, its refinements are (1) {`?x a :Train, :CargoTrain.`}, (2) {`?x a :Train, :PassengerTrain`}, (3) {`?x a :Train; :hasEngine ?y`}.
2. Evaluate patterns (e.g. with some quality measure or using a strategy proposed below) and select only the best ones.
3. Repeat the steps 1-2 as long there are patterns for refinement and maximal number of iterations is not exceeded.

An RDF graph can be simplified to a matrix by *propositionalisation* with Fr-ONT-Qu patterns. Every pattern corresponds to a single column and every individual from the graph to a single row. 1 in the matrix means that given individual is covered by the pattern, and 0 means otherwise.

The detailed description of Fr-ONT-Qu algorithm with proofs of its theoretical properties is presented in (Ławrynowicz and Potoniec 2014).

Ontology enrichment based on mathematical modeling

Fig. 1 illustrates general concept of our method. The method starts from the user-defined root class. It then executes several cycles of refinement and evaluation of class descriptions; each cycle results in a set of more specialized descriptions. In the refinement phase, the refinement operator of Fr-ONT-Qu is applied. In the evaluation phase, a whole *set* of patterns is evaluated with a mathematical model.

In Fr-ONT-Qu's declarative bias, only properties specific for the root class are used (e.g. for the DBpedia class *PopulatedPlace*, the property *province* is used, whereas *highestPlace* is not). Data can sometimes be erroneous, so to denoise generated refinements, a minimal coverage threshold is applied. They are transformed to description logic (DL), which can be easily done thanks to their specific shape.

Our mathematical model is formulated as a linear programming problem, to easily exploit a wide range of preexisting solvers. The goal is to find such a subset of the patterns that maximizes number of individuals covered by an exactly one pattern, which is implemented on a matrix obtained from propositionalisation. At the same time, we require that at least a given number of patterns is selected (usually 2 or 3), to avoid patterns that do not divide the taxonomy. The full model is in the complementary materials[2].

The model is computed during every evaluation phase. Sometimes more than one restriction is required to create a meaningful subclass for a given class, so more than one iteration of Fr-ONT-Qu is required. This creates chains in subsumption hierarchy of the DL classes corresponding to the patterns. To remove these chains, we apply post-processing step by removing classes occurring in the middle of chains. To avoid computationally expensive reasoning services, we use structural subsumption. As Fr-ONT-Qu generates taxonomically-closed patterns, this boils down to checking a subtree-supertree relationship. Formal definition of this process is available in the complementary materials[3].

We did preliminary experiments for a few classes from DBpedia ontology, which generated interesting subhierarchies. They are available in the complementary materials[4].

Conclusions

In this paper, we have presented a method for induction of a branch of the expressive ontology rooted at a given class. We have proposed a novel approach to ontology learning – a combination of class description induction methods with mathematical modeling – to address this task. Our approach is generalizable to arbitrary ontology learning tasks which can be formulated with the use of a mathematical model.

Acknowledgments

Jedrzej Potoniec acknowledges support of Polish National Science Center, grant DEC-2013/11/N/ST6/03065.

References

Bizer, C.; Heath, T.; and Berners-Lee, T. 2009. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.* 5(3):1–22.

Fanizzi, N.; D'Amato, C.; and Esposito, F. 2008. DL-FOIL concept learning in description logics. In *Proc. of the 18th international conference on Inductive Logic Programming*, ILP '08, 107–121. Berlin, Heidelberg: Springer-Verlag.

Ławrynowicz, A., and Potoniec, J. 2011. Fr-ONT: an algorithm for frequent concept mining with formal ontologies. In *Proceedings of the 19th international conference on Foundations of intelligent systems*, 428–437. Springer-Verlag.

Ławrynowicz, A., and Potoniec, J. 2014. Pattern based feature construction in semantic data mining. *Int. J. Semantic Web Inf. Syst.* 10(1):27–65.

Lehmann, J., and Völker, J., eds. 2014. *Perspectives on Ontology Learning*. IOS Press.

Lehmann, J. 2009. DL-Learner: learning concepts in description logics. *J. Mach. Learn. Res.* 10:2639–2642.

Völker, J., and Niepert, M. 2011. Statistical schema induction. In *The semantic web: research and applications - Volume Part I*, 124–138. Berlin, Heidelberg: Springer-Verlag.

[2] semantic.cs.put.poznan.pl/aaai15/m.pdf

[3] semantic.cs.put.poznan.pl/aaai15/sub.pdf

[4] semantic.cs.put.poznan.pl/aaai15/