Learning Multiple Description Logics Concepts

Raphael Melo[1], Kate Revoredo[1], and Aline Paes[2]

[1] Postgraduate Information Systems Program, UNIRIO
Rio de Janeiro, Brazil
[2] Department of Computer Science Institute of Computing, UFF
Rio de Janeiro, Brazil
{raphael.thiago, katerevoredo}@uniriotec.br, alinepaes@ic.uff.br

Abstract. Description logics based languages have became the standard representation scheme for ontologies. They formalize the domain knowledge using interrelated concepts, contained in terminologies. The manual definition of terminologies is an expensive and error prone task, therefore automatic learning methods are a necessity. In this paper we lay the foundations of a multiple concept learning method that uses virtual concepts to aid the learning process, yielding more compact and readable terminologies. In this paper, we define virtual concepts and how they can be implemented in the current concept learning methods. We show through experiments how the method stacks up against other multiple concept learning methods.

1 Introduction

Description logics (DLs) [1] form a family of knowledge representation languages, with different expressive power, that are typically decidable fragments of first order logic (FOL). With DL it is possible to represent domain concepts and their relations. Moreover, due to their computational power and expressiveness, they have been widely used in Semantic Web [2] for ontology representation.

The task of defining the domain knowledge through a DL is usually done manually, which is time consuming and error prone, even more because the domain experts themselves do not always agree about the definitions of concepts and their relationships [3]. Therefore, to consider applying machine learning techniques [4] for automatically learning in DLs is relevant and sometimes even required. A number of DL learning approaches have been proposed in the literature [5] [6]. In these approaches each concept is learned independently from each other, thus, none of the concepts being learned are considered in the definition of the others. If this assumption is removed, the final ontology could be clearer and closer to the way that the concepts are related in the underlying domain. In this sense, another question arises: "What is the best order to learn a set of

[1] The first author would like to thank CAPES for the financial support through a master scholarship.

related concepts?" In [7], we address this problem by defining an algorithm that discovers a taxonomy of the concepts being learned and then uses this taxonomy to define the order for learning the concepts. However, since the order is defined by a heuristic aimed at finding relations between concepts and subconcepts, it may be the case that the ordering found does not yield the best solution for the learning task. Moreover, during the learning process only previously learned concepts are considered.

In this paper, we lay the foundations of multiple concept learning using the ideas discussed in [7] as motivation. Thus, we propose a learning strategy that allows concepts not yet learned to appear in the definition of other concepts, thus making possible to learn more compact and readable terminologies.

The paper is organized as follows. In Section 2, Description Logic and concept learning are reviewed. In Section 3, we present the proposed approach to learn multiple concepts. Section 4 presents some preliminary experimental results. Section 5 concludes the paper and presents the next steps of the research.

2 DLs and concept learning in DLs

DLs knowledge bases ($\mathcal{KB}$) have two components: a *TBox* and an *ABox*. The *TBox* contains intensional knowledge in the form of a terminology. Knowledge is expressed in terms of *individuals*, *concepts*, and *roles*. Thus, the terminology consists of concepts, which denote a set of individuals and roles which denote binary relationships between individuals. In this paper, we assume the common assumption made about DL terminologies: (i) only one definition for a concept name and (ii) concept definitions are acyclic. The *ABox* is composed of assertions about the individuals of the domain. An assertion states that an individual belongs to a concept or that a pair of individuals satisfies a role. Attached to a DL's $\mathcal{KB}$ there must be a reasoning mechanism, responsible for inferring information about individuals from the $\mathcal{KB}$.

There are a number of existing approaches to automatically learn DLs concepts, most of them [5] [6] [8] are inspired by Inductive Logic Programming(ILP) [9] techniques. The goal is to induce concept descriptions from existing evidences. When learning a concept, one has the purpose of finding a generalized and correct definition of such a concept from a set of examples, as defined below:

Definition 1 (Concept Learning)
Given:

- *a knowledge base* $\mathcal{KB}$*,*
- *a target concept* $Target$ *such as* $Target \notin \mathcal{KB}$*,*
- *a set of target examples* $\mathcal{E}$*, divided into positive* ($\mathcal{E}_p$) *and negative* ($\mathcal{E}_n$) *examples, such that* $\mathcal{E} = \mathcal{E}_p \cup \mathcal{E}_n$

Find:

- *A definition of the concept* $C(Target \equiv C)$ *such that* $\mathcal{KB} \cup C \models \mathcal{E}_p$ *and* $\mathcal{KB} \cap C \not\models \mathcal{E}_n$*.*

Although Definition 1 requires that the learned concept covers all the positive examples and none of the negative examples, these hard criteria are usually relaxed to enable the induction.

The concept learning task can be expanded to multiple concept learning, as defined below:

Definition 2 (Multiple Concept Learning)
Given:

- *knowledge base* $\mathcal{KB}$*,*
- n *concept learning tasks* $\{A_{C_1}, A_{C_2}, \ldots, A_{C_n}\}$*, where* $A_{C_i} = \{\mathcal{E}_{p_i}, \mathcal{E}_{n_i}\}$

Find:

- $\mathcal{KB}' = \mathcal{KB} \cup \mathcal{T}$*, where* $\mathcal{T}$ *is a taxonomy which holds the concepts definitions found in the* n *concept learning tasks.*

In this paper we focus on multiple concept learning methods that return a $\mathcal{T}$ that have compact definitions.

In [7] we presented a pre-processing method for multiple concept learning called terminology learning. It yields compact terminologies by defining an order to execute each concept learning task. This order is definied by finding, before the learning process, all the *subsumee* and *subsumer* relationships among the concept learning tasks using the shared individuals in the example sets as evidence.

However, although the *subsumee* and *subsumer* relation is important when devising an order, it is not the only relationship among concepts that can impact the later learned definition. Thus, in this paper we follow a different approach to find out the concepts that should be used to define another concept. Instead of directly finding a taxonomy from the set of examples, we let the learning algorithm decide what is the best way of defining each concept.

3 Multiple DL Concept Learning

The concept learning task can be viewed as a search problem over the space of concepts, created using three basic elements [5]: (i) a refinement operator to build the search tree of concepts; (ii) a search algorithm to control how this search tree is traversed; (iii) a scoring function to evaluate the nodes of the tree and to point out the best current concept candidate.

The refinement operator is responsible for defining a number of rules, from which a valid candidate definition for a concept is yielded. The candidate definitions are created from combinations of *known concepts*, *roles* and *constructors*. The constructors are different according to the DL language chosen. We propose to take into account an additional type of concept when the refinement operator is generating a concept definition, henceforth called *virtual concept*.

Virtual concepts are concepts that do not have an explicit definition yet. As usual, these concepts have a set of examples related to it, divided into positives and negative examples. Once a definition for a concept is learned, it should have considered the individuals marked as positive examples as belonging to it. On the other hand, the negative examples are the individuals that do not belong to that concept and its definition should be able to indicate that. Since each concept has these sets of examples associated to it, it is possible to take into account the set of examples instead of the concept definition when referring to a concept. In this way, we can say that, *while* a concept is not selected to be the target one, it also behaves as a virtual concept.

In order to consider virtual concepts in the concept learning task, it is necessary to make the scoring function cope with them. Usually, the scoring function is either the cover relationship or a variation of it. The cover relationship is a function that evaluates how many positive and negative examples are inferred by a candidate definition. So this can be achieved by transforming the example sets of a virtual concept into assertion inside the ABox, e.g., Albert $\in E_p$ of C, then the assertion C(Albert) will be added to the ABoxes of all concept learning tasks that could use the virtual concept C.

We argue that adding the assertions of a virtual concept can have the same result as that of the regular inference if the following assumptions holds: all the virtual concepts in the multiple concept learning task should share the same ABox and all the relevant individuals for a particular virtual concept should be covered in one of its example sets. If that is indeed the case, it is possible to use virtual concepts in the same way that other non-target instantiated concepts are used inside the concept learning task. Moreover, the addition of virtual concepts in the learning process will allow it to find concept definitions more compact then the ones found with the terminology learning method [7]. The terminology learning method only deals with one type of usage relationship between virtual concepts, the *subsumee* and *subsumer* relationship. This kind of relationship is only related to the concept and subconcept relationship, e.g. in the kinship domain we have $\mathsf{Grandfather} \sqsubseteq \mathsf{Father}$. However, it can not reliably work with relationships that differs from *subsumer* and *subsumee*, for example, the disjunction between Grandfather and Grandmother to define Grandparent. The method proposed in this paper is capable of dealing with it because we turn the learning task into the responsible component for finding relationships between concepts.

The proposed method has the local goal to find the most compact definition for a single virtual concept. The major concern that arises from this is the formation of cycles. This could be avoided by two different approaches: (i) when constructing the ABox' for a target concept, the addition of facts associated to virtual concepts that uses the current target concept in their definition can be avoided. (ii) with a post-processing procedure. The first one is likely to be more efficient in returning a solution, but does not guarantee that this is the optimal solution, while the second may have a better chance to return the optimal solution, but some concepts may be relearned several times, i.e., the learning process can become less efficient.

4 Preliminary Experiments

To evaluate the proposed method some experiments were devised considering the concepts GRANDPARENT (GP), GRANDFATHER (GF) and GRANDMOTHER (GM) from the Family ontology.

A knowledge base of the family domain was used to run the experiments [1]. Each individual cited in the knowledge base is either a positive example or a negative example to a concept. The description learning component chosen to learn the concepts is the DL-Learner system with default settings, since it is a largely used environment to learn DLs.

The first analysis we conducted showed that our proposal, Multiple Concept Learning (MCL), is able to learn a terminology more compact and readable than the Concept Learning (CL) approach, which learns the concepts individually and independently. Table 1 shows the definition for the three concepts. Notice that GRANDPARENT is defined as a disjunction of the two other concepts.

Another evaluation concerns whether MCL is able to learn a more compact terminology when compared with the Terminology Learning (TL) task. For this comparison, we considered two concept orders for MCL:
(i) $< \mathsf{GRANDPARENT}, \mathsf{GRANDFATHER}, \mathsf{GRANDMOTHER} >$ and
(ii) $< \mathsf{GRANDMOTHER}, \mathsf{GRANDFATHER}, \mathsf{GRANDPARENT} >$.
The first one, was found by the approach proposed in [7] and then is the same one used by TL. Moreover, we avoid cycles by changing the correspondent ABox as described in Section 3. The results in Table 1 shows that different learning orders yield different results, and that the proposed method can achieve the same result as the terminology learning by reversing the order. This demonstrates that the proposed method is more versatile than the terminology learning, because it isn't bound to a learning order, while still maintaining the accuracy.

To sum up, these results point out that the method has the potential for finding compact solutions when avoiding cycles with pre-processing (MCL + Order 1 or 2), and its ability to find optimal solutions, if a post processing method to avoid cycle is used (MCL).

5 Conclusion and future remarks

In this paper we laid the foundations over which a multiple concept learning method could be built. We defined a new type of concept, the virtual concept, analogous with the definition of the concept learning task, but with a twist to make it usable inside the existing learning process of another concept. The proposed method opens the possibility of finding all the possible usage relationships among virtual concepts. Because of this, we argued that the use of virtual concepts could yield better results than the ones found with the method presented in [7] and in regular concept learning methods. We also presented directions to deal with the cycle problem that may appear with the use of this method. An

[1] `ftp://ftp.cs.utexas.edu/pub/mooney/forte`

Table 1. Resulting Concepts Definition on All Learning Experiments

Experiment	**Concept**	**Definition**	**Length**
CL	GP	EXISTS parent.EXISTS parent.TOP.	5
	GF	(male AND EXISTS parent.EXISTS parent.TOP).	7
	GM	(female AND EXISTS parent.EXISTS parent.TOP).	7
MCL	GP	(grandfather OR grandmother).	3
	GF	(grandparent AND male).	3
	GM	(grandparent AND female).	3
TL=CL+Order	GP	EXISTS parent.EXISTS parent.TOP.	5
	GF	(grandparent AND male).	3
	GM	(grandparent AND female).	3
MCL+Order 1	GP	(grandfather OR grandmother).	3
	GF	(male AND EXISTS married.grandmother).	5
	GM	(female AND EXISTS parent.EXISTS parent.TOP).	7
MCL+Order 2	GP	EXISTS parent.EXISTS parent.TOP.	5
	GF	(grandparent AND male).	3
	GM	(grandparent AND female).	3

experiment concerning the Kinship domain showed that it is possible to learn a clearer and more compact terminology without requiring a good ordering of the concepts.

In the future we would like to analyze the behavior of the method on different data sets and different DL languages, since we believe the proposed method is capable to work with all possible constructors sets.

References

1. F. Baader and W. Nutt, “Basic description logics,” in *The description logic handbook* (F. Baader, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, eds.), pp. 47–100, Cambridge University Press, 2 ed., may 2010.
2. T. Berners-Lee, J. Hendler, and O. Lassila, “The semantic web,” *Scientific American*, vol. 5, no. 284, p. 34, 2001.
3. A. Maedche and S. Staab, “Ontology learning for the semantic web,” *IEEE Intelligent Systems and Their Applications*, vol. 16, no. 2, pp. 72–79, 2001.
4. T. Mitchell, *Machine Learning.* New York: McGraw-Hill, 1997.
5. J. Lehmann and P. Hitzler, “Concept learning in description logics using refinement operators,” *Machine Learning*, vol. 78, no. 1-2, pp. 203–250, 2010.
6. N. Fanizzi, C. d’Amato, and F. Esposito, “Dl-foil concept learning in description logics,” in *Proceedings of the 18th International Conference on Inductive Logic Programming (ILP-2008)*, vol. 5194 LNAI of *Lecture Notes in Computer Science*, pp. 107–121, Springer, 2008.
7. R. Melo, K. Revoredo, and A. Paes, “Terminology learning through taxonomy discovery,” *BRACIS ”to appear”*, 2013.
8. L. Iannone, I. Palmisano, and N. Fanizzi, “An algorithm based on counterfactuals for concept learning in the semantic web,” *Applied Intelligence*, vol. 26, no. 2, pp. 139–159, 2007.
9. L. De Raedt, *Logical and Relational Learning.* Springer, 2008.