An Inductive Approach to Assertional Mining for Web Ontology Revision

Chieko Nakabasami

Toyo University, 1-1-1 Izumino Itakura Oura Gunma 374-0193, Japan
chiekon@itakura.toyo.ac.jp

Abstract. This paper proposes an inductive learning method for maintaining a web-based ontology by incorporating newly generated concepts from assertional knowledge (A-Box). The ontology used in this approach is represented by DAML+OIL. This ontology is translated into a form acceptable for the FACT system, a Description Logic (DL) reasoner, and is compiled into a knowledge base as a T-Box, a terminological knowledge description. Inductive learning is used for integrating the A-Box, where positive and negative examples submitted by human users are stored. Inductive Logic Programming (ILP) is used in order to induce concepts consistent with positive examples and to exclude negative ones. Such induced concepts are explored in order to find where they are positioned in the concept hierarchy in the T-Box, and the original ontology is revised. ILP can provide new concepts for DLs even though they may have richer expressiveness since DL is a decidable fragment of first-order logic. The induced concepts could be also utilized for predicting novel assertions from human users.

1 Introduction

In recent years, the Semantic Web[23] has been evolving as the next-generation web technology and has attracted the attention of many researchers in machine learning and knowledge engineering. The Semantic Web opens a wide range of new research challenges for the machine-learning community. Ontologies play a key role in the Semantic Web, which relies significantly on the formal ontologies that structure underlying data for the purpose of comprehensive and transportable machine understanding[15]. In learning ontology for the Semantic Web, the idea of automatically maintaining ontologies by analyzing instance data, which has recently been called A-Box-Mining[16], is not new, as denoted in [14]. In regard to A-Box Mining research, Assertional Mining in [22] and the rough set theory [18] have been adopted for assertional mining where each concept has been described in Description Logics (DL). Wellington[28], which can check the consistency of an A-Box and was developed by a group at King's College, has been released on a web site. DL reasoning service is used for information extraction by checking new concept descriptions and adding them to the domain ontology[25]. On the other hand, as a research of applying ILP to knowledge base represented in DL[1], the main purpose of which was that T-Box, i.e., terminological knowledge or intentional concept definition, was revised so that entire T-Box knowledge could be consistent. This paper proposes an Inductive Logic Programming method for A-Box-Mining, and generated concepts resulting from mining are put in a suitable position in the ontology constructed in the T-Box concept hierarchy. ILP has been applied to a number of concept inference problems; however, few studies have been conducted with the goal of inducing general concept definitions of knowledge on the WWW, and the Semantic Web in particular. In this paper, DAML+OIL[7] is chosen as a web-based ontology representing language based on DL[13]. The FACT system[9] is used for the inference engine for DL reasoning. First, the DAML+OIL-based ontology is compiled and stored as knowledge bases in processable forms by FACT. Then extensional concepts provided by human users are collected via a Web browser where each concept example is labeled as positive or negative. These examples are regarded as A-Box knowledge, and Aleph[24], an ILP system, is applied to induce general concepts that are consistent with all the positive data and inconsistent with all the negative data. Such generated concepts are put into the DAML+OIL ontology hierarchy described above in an appropriate position. The DAML+OIL ontology could be maintained and revised for emerging new concepts submitted by users. In addition, in the A-Box, the induced concept could predict positive or negative concepts against newly incorporated users' assertions. This paper is organized as follows: a brief introduction of DL is given in Section 2. Section 3 explains how DAML+OIL ontology is transformed into understandable forms by FACT. Section 4 shows a small example and reports a result. The conclusion is given in Section 5.

2 Description Logics

Description Logics (DL) are logic-based-knowledge representation formalisms, also known as terminological logics or concept languages based on concepts (classes) and roles (e.g., [3][17]). Concepts are interpreted as sets of objects and roles as binary relations of objects. DLs are characterized by sets of constructors provided for building complex concepts and roles from simpler ones. The basic DL is known as $\mathcal{ALC}$, in which concepts (denoted by C, D) are constructed out of atomic concepts (denoted by A) and atomic roles (denoted by P) according to the following syntax rules and semantics (Table 1). In Table 1, semantics is given by an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, which consists of a set $\Delta^{\mathcal{I}}$ (the domain of $\mathcal{I}$) and a function $\cdot^{\mathcal{I}}$ (the interpretation function of $\mathcal{I}$) that maps every concept to a subset of $\Delta^{\mathcal{I}}$ and every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

In DL, the knowledge base consists of two knowledge parts: the T-Box and the A-Box. The T-Box, the terminological part, is a set of axioms describing the domain structure, where concepts and relations holding between such concepts are defined. In the T-Box, terminological axioms are restricted to formulas of the form $C \sqsubseteq D$ (short for $\neg C \sqcup D$) and $C \doteq D$ (short for $(\neg C \sqcup D) \sqcap (C \sqcup \neg D)$), where C and D are concept names. Given an interpretation $\mathcal{I}$ which satisfies $C \doteq D$ iff $C^{\mathcal{I}} = D^{\mathcal{I}}$ and $C \sqsubseteq D$ iff $C^{I} \subseteq D^{I}$, a T-Box $\mathcal{T}$ is consistent iff it

Table 1. DL syntax rules and semantics

Constructor	Syntax	Example	Semantics
atomic concept	A	Human	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
atomic role	R	has-child	$R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	Human $\sqcap$ Male	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	Doctor $\sqcup$ Lawyer	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
negation	$\neg C$	$\neg$Male	$\Delta^{\mathcal{I}} \backslash C$
existential restriction	$\exists R.C$	$\exists$has-child.Male	$\{x \mid \exists y.\langle x, y\rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
value restriction	$\forall R.C$	$\forall$has-child.Doctor	$\{x \mid \forall y.\langle x, y\rangle \in R^{\mathcal{I}} \Longrightarrow y \in C^{\mathcal{I}}\}$

satisfies every axiom in $\mathcal{T}$ (i.e., $\mathcal{I} \models \mathcal{T}$). On the other hand, the A-Box, the assertional part, is a set of axioms describing a concrete situation. It consists of concept assertions (written as $a : C$) and role assertions (written as $\langle a, b\rangle : R$). Given an interpretation $\mathcal{I}$ which satisfies $a : C$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $\langle a, b\rangle : R$ iff $\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \in R^{\mathcal{I}}$, an A-Box $\mathcal{A}$ is consistent if it satisfies every axiom in $\mathcal{A}$ (i.e., $\mathcal{I} \models \mathcal{A}$). A knowledge base $\Sigma = \langle \mathcal{T}, \mathcal{A}\rangle$ iff it satisfies both $\mathcal{T}$ and $\mathcal{A}$ ($\mathcal{I} \models \Sigma$). DLs provide several reasoning services such as concept satisfiability, concept subsumption, knowledge base consistency, and instance checking. The tableaux algorithm is used for such reasoning[4].

3 DAML+OIL-based Knowledge Base in FACT

3.1 Overview of DAML+OIL

DAML+OIL is a semantic markup language for Web resources based on DL. It is regarded as a T-Box in the sense that it describes structures of the domain. It builds on earlier W3C[29] standards such as RDF[20] and RDF Schema[21] and extends these languages with richer modeling primitives. Class constructors and axioms equipped for DAML+OIL[10] are shown in Table 2and Table 3. In addition to these denotations, in DAML+OIL, XMLS[26] data types are supported, and the arbitrarily complex nesting of constructors is allowed: e.g., $\forall$hasChild.(Doctor $\sqcup$ $\exists$hasChild.Doctor), which means "individuals whose children are all doctors or individuals who have at least one child who is a doctor." Note that the axioms denoted in Table 3 are mostly reducible to subClassOf or subPropertyOf axioms.

Table 2. DAML+OIL class constructors

Constructor	DL Syntax	Example
intersectionOf	$C_1 \sqcap \cdots \sqcap C_n$	Human $\sqcap$ Male
unionOf	$C_1 \sqcup \cdots \sqcup C_n$	Doctor $\sqcup$ Lawyer
complementOf	$\neg C$	$\neg$Male
oneOf	$\{x_1 \cdots x_n\}$	{john,Mary}
toClass	$\forall P.C$	$\forall$hasChild.Doctor
hasClass	$\exists P.C$	$\exists$hasChild.Lawyer
hasValue	$\exists P.\{x\}$	$\exists$citizenOf.{USA}
minCardinalityQ	$\geqslant nP.C$	$\geqslant$2hasChild.Lawyer
maxCardinalityQ	$\leqslant nP.C$	$\leqslant$1hasChild.Male
cardinalityQ	$= nP.C$	=1hasParent.Female

Table 3. DAML+OIL axioms

Axiom	DL Syntax	Example
subClassOf	$C_1 \sqsubseteq C_2$	Human $\sqsubseteq$ Animal $\sqcap$ Biped
sameClassAs	$C_1 \doteq C_2$	Man $\doteq$ Human $\sqcap$ Male
subPropertyOf	$P_1 \sqsubseteq P_2$	hasDaughter $\sqsubseteq$ hasChild
samePropertyAs	$P_1 \doteq P_2$	cost $\doteq$ price
sameIndividualAs	$\{x_1\} \doteq \{x_2\}$	President_Bush $\doteq$ G_W_Bush
disjointWith	$C_1 \sqsubseteq \neg C_2$	Male $\sqsubseteq$ $\neg$Female
differentInduvidualFrom	$\{x_1\} \sqsubseteq \neg\{x_2\}$	John $\sqsubseteq$ $\neg$Peter
inverseOf	$P_1 \doteq P_2^-$	hasChild $\doteq$ $\neg$hasParent$^-$
transitiveProperty	$P^+ \sqsubseteq P$	ancester$^+$ $\sqsubseteq$ ancester
uniqueProperty	$\top \sqsubseteq \leqslant 1P$	$\top \sqsubseteq \leqslant$ 1 hasMother
UnambiguousProperty	$\top \sqsubseteq \leqslant 1P^-$	$\top \sqsubseteq \leqslant$ 1 isMotherOf$^-$

For example, a fragment of a DAML+OIL ontology description[2] is written in Figure 1.The first description asserts that there is a class known as “Animal,” and the second means that “Male” is a subclass of “Animal.” The last means some animals are “Female” but nothing can be both “Male” and “Female” because these two classes are disjoint.

```
<daml:Class rdf:ID="Animal"></daml:Class>
<daml:Class rdf:ID="Male">
   <rdfs:subClassOf rdf:resource="#Animal"/>
</daml:Class>
<daml:Class rdf:ID="Female">
   <rdfs:subClassOf rdf:resource="#Animal"/>
<daml:disjointWith rdf:resource="#Male"/>
</daml:Class>
```

Fig. 1. DAML+OIL sample description

3.2 Translation from DAML+OIL Ontology to FACT Knowledge Base

This paper focuses on the ontology concerning universities[11], which was obtained from the DAML Ontology Library site[8]. It describes universities and the activities that occur at them: e.g., Professor, Assistant, UndergraduateStudent as classes, and mastersDegreeFrom, teacherOf as properties. A fragment of the university ontology is illustrated in Figure 2.

```
<Class ID="TeachingAssistant">
   <label>university teaching assistant</label>
   <subClassOf resource="#Assistant" />
</Class>
<Property ID="teachingAssistantOf">
   <label>is a teaching assistant for</label>
   <domain resource="#TeachingAssistant" />
   <range resource="#Course" />
</Property>
```

Fig. 2. A fragment of the University Ontology

The university ontology written in DAML+OIL is translated into the FACT knowledge base. FACT is a DL classifier which includes two reasoners, one for the logic $\mathcal{SHF}$ ($\mathcal{ALC}$ augmented with transitive roles, functional roles, and role hierarchy) and the other for the logic $\mathcal{SHIQ}$ ($\mathcal{SHF}$ augmented with inverse

Table 4. XML concept descriptions

Standard Notation	XML Markup
$\top$	<TOP/>
$P_1 \sqcap \neg P_2$	<AND> <PRIMITIVE NAME="P1"/> <NOT> <PRIMITIVE NAME="P2"/> </NOT> </AND>
$\exists R.P$	<EXISTS> <PRIMROLE NAME="R"/> <PRIMITIVE NAME="P"/> </EXISTS>

roles and a qualified number restriction)[9]. In FACT, both reasoners are decidable because sound and complete tableaux algorithms are presented [9][12]. In order to implement FACT, a server on which the FACT server is running is used, and client applications can access the server via a CORBA interface independent of their architecture[5]. DAML+OIL is regarded as $\mathcal{SHIQ}$ plus nominals and datatypes with RDFS-based syntax, so DL reasoning can be used with DAML+OIL on FACT. In FACT, concept descriptions in the knowledge base are represented in the XML format. For example, some concept descriptions are represented as shown in Table 4.

The university ontology is translated from the DAML+OIL format into the above XML by means of XSLT[27]. For example, the university ontology shown in Figure 2 is translated into XML in Figure 3.

```
<KNOWLEDGEBASE>
<DEFCONCEPT NAME="TeachingAssistant" />
<IMPLIESC>
<CONCEPT>
<PRIMITIVE NAME="TeachingAssistant" />
</CONCEPT>
<CONCEPT>
<PRIMITIVE NAME="Assistant" />
</CONCEPT>
</IMPLIESC>
<DEFROLE NAME="teachingAssistantOf" />
<IMPLIESC>
<CONCEPT>
<SOME>
<PRIMROLE NAME="teachingAssistantOf" />
<TOP/>
</SOME>
</CONCEPT>
<CONCEPT>
<PRIMITIVE NAME="TeachingAssistant" />
</CONCEPT>
</IMPLIESC>
<IMPLIESC>
<CONCEPT>
<TOP/>
</CONCEPT>
<CONCEPT>
<ALL>
<PRIMROLE NAME="teachingAssistantOf" />
<PRIMITIVE NAME="Course" />
</ALL>
</CONCEPT>
</IMPLIESC>
</KNOWLEDGEBASE>
```

Fig. 3. XML representation translated from the University Ontology with DAML+OIL

The description in Figure 3 is equivalent to the following denotation in DL:

$$\begin{array}{l}
\mathsf{TeachingAssistant} \sqsubseteq \mathsf{Assistant} \\
\exists \mathsf{teachingAssistantOf}.\top \sqsubseteq \mathsf{TeachingAssistant} \\
\forall \mathsf{teachingAssistantOf}.\mathsf{Course}
\end{array}$$

According to the DTD for the FACT knowledge base[5], the university ontology is translated into an XML document. There are 49 concepts and 26 roles in the ontology. The XML document is compiled into FACT and provided as a T-Box for the university ontology.

4 Introductory Example

Some examples are prepared for an A-Box, which consists of the descriptions inputted by humans. As a target concept, “Doctor Course Student” is intended. Two positive examples and one negative example are supposed, as shown in Figure 4. In this figure, each concept description for the target concept should have more than two examples, but for the sake of simplicity, the rest have been eliminated. Note that for the third concept (concept 3) no description of “mastersDegreeFrom” exists.

```
% concept 1 (positive)
mastersDegreeFrom(c1,univ1).
teachingAssistantOf(c1,computerScience).

% concept 2 (positive)
researchProject(c2,machineLearning).
mastersDegreeFrom(c2,univ2).

% concept 3 (negative)
doctralDegreeFrom(c3,univ1).
professor(c3).
```

Fig. 4. Positive and negative examples

Aleph is used for inducing general concepts that are consistent with the positive examples and inconsistent with the negative. The background knowledge for Aleph is tailored along the university ontology. Aleph induces the following concept:

$$mastersDegreeFrom(X, Y).$$

By refering to the type definition in Aleph’s background knowledge, the above concept could be transformed into the following concept with DL.

$$\exists \mathsf{mastersDegreeFrom.University}$$

Then the induced concept is inputted into the university T-Box. FACT provides a function named “taxonomy_position,” which receives concept descriptions as arguments and returns the position in the taxonomy of the given concept expression; i.e., the super-concept for the given concept, the sub-concepts, and the equivalent ones. The above induced concept accepts the following position in the university T-Box:

$$\begin{array}{l} \mathsf{super : person} \\ \mathsf{sub} : \bot \\ \mathsf{equivalent : none} \end{array}$$

As a result, the target concept “Doctor Course Student” is newly created and is put between “person” and “bottom” (i.e., the concept becomes a leaf) in the university T-Box.

5 Conclusion

In this paper, we have presented a method for revising web-based ontology by applying ILP to A-Box mining. [22] proposed a learning method for the A-Box by applying the rough set theory. In the framework of [22], one simple description logic $\mathcal{AL}^{\mathbb{R},=}$ is used for learning: their framework receives an A-Box and a set of decision concepts, called a D-Box, and outputs the set of all generalized decision concepts (GDCs) for each decision in the D-Box by applying efficient and reliable algorithms in the rough set theory. There exists a simple translation of the algorithms used for data mining with the rough set theory when concepts can be described by $\mathcal{AL}^{\mathbb{R},=}$; however, in the case of more expressive A-Box languages, no algorithm exists, and the complexity of calculating GDCs becomes an issue. A way of defining criteria to stop algorithms is suggested for solving such problems. On the other hand, Horn Clauses are usually used in ILP. Since DLs are subsets of function-free first-order logic and a decidable fragment of it, the DL language with DAML+OIL can be treated in ILP only if the A-Box is carefully constructed to be function-free and with at most two variables. The main contribution of this paper is the incorporation of ILP into a web-based ontology based on DL in order to facilitate the revision of a relatively expressible knowledge representation. From the machine learning point of view, [6] applied relational learning to construct knowledge bases from the web. In [6], FOIL[19] is used for the induction of intentional knowledge from the web resources regarded as extensional knowledge of the specified domain. The Semantic Web has been advocated by the WWW Consortium[29] and could play a critical role in the semantic phase of the next generation of web resources. Web resources could gain a primary position in the sense that they have data from which various types of knowledge are extracted. DAML+OIL is thought to be one of the main ontology formalisms, and it is important for web applications and services to revise and maintain ontologies constructed from it. The rich expressiveness which ILP possesses is one of the promising methods for inferring general concepts and constructing ontologies on the web.

References

1. Alvarrez, J.: A Formal Framework for Theory Learning using Description Logics. Intl. Workshop on Inductive Logic Programming (ILP’00), Work in Progress track, London. (2000)
2. Annotated DAML+OIL (March 2001) Ontology Markup. (2001) http://www.daml.org/2001/03/daml+oil-walkthru.html.

3. Baader, F., Hollunder, B.: A terminological Knowledge Representation Systems with Complete Algorithm. Proc. of the 1st Intl. Workshop on Processing Declarative Knowledge, Lecture Notes in Computer Science 572. Springer-Verlag. (1991) 67–85
4. Baader, F., Sattler, U.: Tableaux Algorithms for Description Logics. Automated Reasoning with Tableaux and Related Methods, Proc. of Tableaux 2000, No.1847 in LNAI, Springer Verlag. (2000) 1–18
5. Bechhofer, S., Horrocks, I., Tessaris, S.: CORBA Interface for a DL Classifier. (1999)
6. Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S.: Learning to Construct Knowledge Bases from the World Wide Web. Artificial Intelligence, 118(1-2). (2000) 69–113.
7. DAML Language: http://www.daml.org/language/
8. DAML Ontology Library: http://www.daml.org/ontologies/
9. The Fact System: http://www.cs.man.ac.uk/fact
10. Fikes, R., McGuinness, D.: An Axiomatic Semantics for RDF, RDF-S, and DAML+OIL.
http://www.w3.org/TR/daml+oil-axioms, March 2001. (2001)
11. Heflin, J.: university-ont.
http://www.cs.umd.edu/projects/plus/DAML/onts/univ1.0.daml
12. Horrocks, I.: Using an Expressive Description Logics: Fact or Fiction?. Principles of Knowledge Representation and Reasoning: Proc. of the 6th Intl.Conf.(KR'98) Morgan Kaufmann. (1998) 636–647
13. Horrocks, I.: DAML+OIL and Description Logic Reasoning. (2001)
http://www.cs.man.ac.uk/ horrocks/Slides/hp-labs.pdf
14. Kietz, J. U., Morik, K.: A Polynomial Approach to the Constructive Induction of Structural Knowledge. Machine Learning Journal,14(2). (1994) 193–218
15. Maedche, A.: A Machine Learning Perspective for the Semantic Web. Position Paper. The 1st Semantic Web Working Symposium(SWWS). California, USA. (2001).
16. Maedche, A., Staab, S.: Learning Ontologies for the Semantic Web. IEEE Intelligent Systems, 16(2), Special Issue on Semantic Web. (2001)
17. Patel-Schneider, P., Swartout, B.: Description-Logic Knowledge Representation System Specification from the KRSS Group of the ARPA Knowledge Sharing Effort. Technical Report, DARPA Knowledge Representation System Specification (KRSS) Group of the Knowledge Sharing Initiative. (1993)
18. Pawlak, Z.: Rough Sets. Intl. Journal of Computer and Information Sciences, 11(5). (1982) 341–156
19. Quinlan, J. R., Cameron-Jones, R. M.: FOIL: A Midterm Report. Proc. of the European Conference on Machine Learning. (1993) 3–20
20. Resource Description Framework. http://www.w3.org/RDF/
21. Resource Description Framework Schema. http://www.w3.org/TR/2000/CR-rdf-schema-20000327/
22. Schlobach, S.: Assertional Mining in Description Logics. Intl. Workshop on Description Logics, CEUR Workshop Proceedings, volume 33. (2000)
23. Semantic Web Portal: http://www.semanticweb.org/
24. Srinivasan, A., Camacho, R.: The Aleph Manual. (1993)
http://www.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/
25. Todirascu, A., Beuvron, F., Galea, D., Rousselot, F.: Using Description Logics for Ontology Extraction. Proc. of ROMAND'2000 Workshop on Robust Parsing. (2001) 89–105
26. XML Scheme. http://www.w3.org/XML/Schema
27. XSL Transformations (XSLT) Version 1.0. http://www.w3.org/TR/xslt
28. Wellington 1.0. http://www.dcs.kcl.ac.uk/research/groups/logics/wellington/
29. WWW Consortium. http://www.w3.org/