Approximating Horn Knowledge Bases in Regular Description Logics to Have PTIME Data Complexity

Linh Anh Nguyen

Institute of Informatics, University of Warsaw
ul. Banacha 2, 02-097 Warsaw, Poland
nguyen@mimuw.edu.pl

This work is a continuation of our previous works [4,5]. We assume that the reader is familiar with description logics (DLs). A knowledge base in a description logic is a tuple $(\mathcal{R},\mathcal{T},\mathcal{A})$ consisting of an RBox $\mathcal{R}$ of assertions about roles, a TBox $\mathcal{T}$ of global assumptions about concepts, and an ABox $\mathcal{A}$ of facts about individuals (objects) and roles. The instance checking problem in a DL is to check whether a given individual a is an instance of a concept C w.r.t. a knowledge base $(\mathcal{R},\mathcal{T},\mathcal{A})$, written as $(\mathcal{R},\mathcal{T},\mathcal{A}) \models C(a)$. This problem in DLs including the basic description logic $\mathcal{ALC}$ (with $\mathcal{R} = \emptyset$) is EXPTIME-hard. From the point of view of deductive databases, $\mathcal{A}$ is assumed to be much larger than $\mathcal{R}$ and $\mathcal{T}$, and it makes sense to consider the data complexity, which is measured when the query consisting of $\mathcal{R}$, $\mathcal{T}$, C, a is fixed while $\mathcal{A}$ varies as input data. It is desirable to find and study fragments of DLs with PTIME data complexity. Several authors have recently introduced a number of Horn fragments of DLs with PTIME data complexity [2,1,3]. The most expressive fragment from those is Horn-$\mathcal{SHIQ}$ introduced by Hustadt et al. [3]. It assumes, however, that the constructor $\forall R.C$ does not occur in bodies of program clauses and goals. The data complexity of the "general Horn fragment of $\mathcal{ALC}$" is coNP-hard [6]. So, to obtain PTIME data complexity one has to adopt some restrictions for the "general Horn fragments of DLs". The goal is to find as less restrictive conditions as possible.

A RBox is a finite set of assertions of the form $R_{s_1} \circ \ldots \circ R_{s_k} \sqsubseteq R_t$, where $R_{s_1}, \ldots, R_{s_k}, R_t$ are role names. A regular RBox is an RBox whose set of corresponding grammar rules $t \to s_1 \ldots s_k$ forms a grammar such that the set of words derivable from any symbol s using the grammar is a regular language specified by a finite automaton. We assume that the corresponding finite automata specifying $\mathcal{R}$ are given when $\mathcal{R}$ is considered. By $\mathcal{R}eg$ we denote $\mathcal{ALC}$ extended with regular RBoxes. We extend the language of $\mathcal{ALC}$ and $\mathcal{R}eg$ with the concept constructor $\forall\exists$, which creates a concept $\forall\exists R_t.C$ from a role name R_t and a concept C. Let $\mathsf{Sem}_1(\forall\exists R_t.C) = \{\forall R_t.C, \exists R_t.\top\}$ and $\mathsf{Sem}_{2,\mathcal{R}}(\forall\exists R_t.C) = \{\forall R_t.C\} \cup \{\forall R_{s_1} \ldots \forall R_{s_{i-1}} \exists R_{s_i}.\top \mid R_{s_1} \circ \cdots \circ R_{s_k} \sqsubseteq R_t$ is a consequence of $\mathcal{R}$ and $1 \leq i \leq k\}$. Then, for a model $\mathcal{I} = \langle \Delta^\mathcal{I}, \cdot^\mathcal{I} \rangle$ of an RBox $\mathcal{R}$, define $x \in \Delta^\mathcal{I}$ to be an instance of a concept C in $\mathcal{I}$ w.r.t. $\mathfrak{s} \in \{\mathsf{Sem}_1, \mathsf{Sem}_{2,\mathcal{R}}\}$, write $\mathcal{I}, x \models_\mathfrak{s} C$, in the usual way if C is not of the form $\forall\exists R_t.D$, and that $\mathcal{I}, x \models_\mathfrak{s} \forall\exists R_t.D$ if $\mathcal{I}, x \models_\mathfrak{s} D'$ for every $D' \in \mathfrak{s}(\forall\exists R_t.D)$. We write $\mathcal{I} \models_\mathfrak{s} C(a)$ for $\mathcal{I}, a^\mathcal{I} \models_\mathfrak{s} C$.

A *positive concept* is a concept (in the extended language) without the constructors $\bot$, $\neg$, $\sqsubseteq$, $\doteq$. A *deterministic positive concept* is a positive concept which does not contain the constructor $\forall$ (but may contain $\exists$ and $\forall\exists$).

V. Dahl and I. Niemelä (Eds.): ICLP 2007, LNCS 4670, pp. 438–439, 2007.

A *positive logic program* is a finite set of *program clauses* formed using the following BNF grammar, in which C denotes a positive concept without $\forall\exists$:

$$D ::= \top \mid A \mid C \sqsubseteq D \mid D \sqcap D \mid \forall R_t.D \mid \exists R_t.D$$

A *deterministic positive logic program* is a finite set of *deterministic program clauses* formed using the above BNF grammar with C being a deterministic positive concept.

Given a positive logic program $\mathcal{T}'$, the deterministic version of $\mathcal{T}'$ is the deterministic positive logic program $\mathcal{T}$ obtained from $\mathcal{T}'$ by replacing every concept of the form $\forall R.C$ in bodies of program clauses of $\mathcal{T}'$ by $\forall\exists R.C$.

In the full version [6] of this paper, we present an algorithm that, given $\mathfrak{s} \in \{\mathsf{Sem}_1, \mathsf{Sem}_{2,\mathcal{R}}\}$ and a knowledge base $(\mathcal{R}, \mathcal{T}, \mathcal{A})$ consisting of a regular RBox $\mathcal{R}$, a deterministic positive logic program $\mathcal{T}$, and an ABox $\mathcal{A}$, constructs a finite "least $\mathfrak{s}$-pseudo-model" I of $(\mathcal{R}, \mathcal{T}, \mathcal{A})$. For $\mathfrak{s} = \mathsf{Sem}_{2,\mathcal{R}}$, the least $\mathfrak{s}$-pseudo-model I has the property that, $I \models_{\mathsf{Sem}_{2,\mathcal{R}}} C(a)$ iff $(\mathcal{R}, \mathcal{T}, \mathcal{A}) \models_{\mathsf{Sem}_{2,\mathcal{R}}} C(a)$, for every deterministic positive concept C. For $\mathfrak{s} = \mathsf{Sem}_1$, we have only the one-way assertion: $I \models_{\mathsf{Sem}_1} C(a)$ implies $(\mathcal{R}, \mathcal{T}, \mathcal{A}) \models_{\mathsf{Sem}_1} C(a)$, for every positive concept C.

Given a Horn knowledge base $(\mathcal{R}, \mathcal{T}', \mathcal{A})$ and a positive concept C, where $\mathcal{R}$ is a regular RBox and $\mathcal{T}$ is a positive logic program, we propose the approximation of checking $(\mathcal{R}, \mathcal{T}', \mathcal{A}) \models C(a)$ by checking whether $I \models_{\mathsf{Sem}_1} C(a)$, where $\mathcal{T}$ is the deterministic version of $\mathcal{T}'$ and I is the least Sem_1-pseudo-model of $(\mathcal{R}, \mathcal{T}, \mathcal{A})$ constructed by the algorithm given in [6]. The approximation is correct in the sense that $I \models_{\mathsf{Sem}_1} C(a)$ implies $(\mathcal{R}, \mathcal{T}', \mathcal{A}) \models C(a)$. It is a good approximation because that:

- First, Sem_1, which interprets $\forall\exists R.D$ in premises as $\forall R.D \sqcap \exists R.\top$, is highly "acceptable". For example, $\forall\exists child.good\_person \sqsubseteq happy\_parent$ w.r.t. Sem_1 is more "acceptable" than the clause $\forall child.good\_person \sqsubseteq happy\_parent$.
- Second, $(\mathcal{R}, \mathcal{T}, \mathcal{A}) \models_{\mathsf{Sem}_{2,\mathcal{R}}} C(a)$ implies $I_1 \models_{\mathsf{Sem}_1} C(a)$ if C is a deterministic positive concept. At least, when the constructors $\forall R.D$ and $\forall\exists R.D$ are disallowed in C and bodies of program clauses of $\mathcal{T}'$ (as assumed for Horn-$\mathcal{SHIQ}$ [3]), the approximation is exact, i.e. $(\mathcal{R}, \mathcal{T}', \mathcal{A}) \models C(a)$ iff $I \models_{\mathsf{Sem}_1} C(a)$.
- Third, I is constructed in polynomial time in the size of $\mathcal{A}$, and checking $I \models_{\mathsf{Sem}_1} C(a)$ can be done in polynomial time in the size of I and C. That is, the approximation has PTIME data complexity.

References

1. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Data complexity of query answering in description logics. In: Proc. of KR'2006, pp. 260–270. AAAI Press, Stanford (2006)
2. Grosof, B.N., Horrocks, I., Volz, R., Decker, S.: Description logic programs: combining logic programs with description logic. In: Proceedings of WWW'2003, pp. 48–57 (2003)
3. Hustadt, U., Motik, B., Sattler, U.: Data complexity of reasoning in very expressive description logics. In: Proceedings of IJCAI-05, pp. 466–471. Professional Book Center (2005)
4. Nguyen, L.A: A bottom-up method for the deterministic Horn fragment of the description logic $\mathcal{ALC}$. In: Proceedings of JELIA 2006, LNAI 4160, pp. 346–358. Springer, Heidelberg (2006)
5. Nguyen, L.A.: On the deterministic Horn fragment of test-free PDL. In: Advances in Modal Logic - vol. 6, pp. 373-392. King's College Publications (2006)
6. Nguyen, L.A.: The full version of this paper, `http://www.mimuw.edu.pl/~nguyen/RG.pdf`