Ontology Learning from Interpretations in Lightweight Description Logics

Szymon Klarman[1,2] and Katarina Britz[2,3]

[1] Department of Computer Science, Brunel University London
[2] CSIR Centre for Artificial Intelligence Research, South Africa
[3] Department of Information Science, Stellenbosch University

Abstract. Data-driven elicitation of ontologies from structured data is a well-recognized knowledge acquisition bottleneck. The development of efficient techniques for (semi-)automating this task is therefore practically vital — yet, hindered by the lack of robust theoretical foundations. In this paper, we study the problem of learning Description Logic TBoxes from interpretations, which naturally translates to the task of ontology learning from data. In the presented framework, the learner is provided with a set of positive interpretations (i.e., logical models) of the TBox adopted by the teacher. The goal is to correctly identify the TBox given this input. We characterize the key constraints on the models that warrant finite learnability of TBoxes expressed in selected fragments of the Description Logic $\mathcal{EL}$ and define corresponding learning algorithms.

1 Introduction

In the advent of the Web of Data and various "e-" initiatives, such as e-science, e-health, e-governance, etc., the focus of the classical knowledge acquisition bottleneck becomes ever more concentrated around the problem of constructing rich and accurate ontologies enabling efficient management of the existing abundance of data [1]. Whereas the traditional understanding of this bottleneck has been associated with the necessity of developing ontologies *ex ante*, in a top-down, data-agnostic manner, this seems to be currently evolving into a new position, recently dubbed the knowledge reengineering bottleneck [2]. In this view, the contemporary challenge is to, conversely, enable data-driven approaches to ontology design — methods that can make use and make sense of the existing data, be it readily available on the web or crowdsourced, leading to elicitation of the ontological commitments implicitly present on the data-level. Even though the development of such techniques and tools, which could help (semi-)automate thus characterized ontology learning processes, becomes vital in practice, the robust theoretical foundations for the problem are still rather limited. This work is an attempt at establishing exactly such foundations and focuses on some key theoretical issues towards this goal.

This work was funded in part by the National Research Foundation under Grant no. 85482.

We study the problem of learning *Description Logic* (DL) TBoxes from interpretations, which naturally translates to the task of ontology learning from data. DLs are a popular family of knowledge representation formalisms [3], which have risen to prominence as, among others, the logics underpinning different profiles of the Web Ontology Language OWL[4]. In this paper, we focus on the lightweight DL $\mathcal{EL}$ [4] and some of its more specific fragments. This choice is motivated, on the one hand, by the interesting applications of $\mathcal{EL}$, especially as the logic behind OWL 2 $\mathcal{EL}$ profile, while on the other, by its relative complexity, which enables us to make interesting observations from the learning perspective. Our learning model is a variant of learning from positive interpretations (i.e., from models of the target theory) — a generally established framework in the field of inductive logic programming [5,6]. In our scenario, the goal of the learner is to correctly identify the target TBox $\mathcal{T}$ given a finite set of its finite models. Our overarching interest lies in algorithms warranting effective learnability in such setting with no or minimum supervision. Our key research questions and contributions are therefore concerned with the identification of specific languages and conditions on the learning input under which such algorithms can be in principle defined.

In the following two sections, we introduce DL preliminaries and discuss the adopted learning model. In Section 4, we identify two interesting fragments of $\mathcal{EL}$, called $\mathcal{EL}^{\mathrm{rhs}}$ and $\mathcal{EL}^{\mathrm{lhs}}$, which satisfy some basic necessary conditions enabling finite learnability, and at the same time, we show that full $\mathcal{EL}$ does not meet that same requirement. In Section 5, we devise a generic algorithm which correctly identifies $\mathcal{EL}^{\mathrm{rhs}}$ and $\mathcal{EL}^{\mathrm{lhs}}$ TBoxes from finite data, employing a basic equivalence oracle. Further, in case of $\mathcal{EL}^{\mathrm{rhs}}$, we significantly strengthen this result by defining an algorithm which makes no such calls to an oracle, and thus supports fully unsupervised learning. In Section 6, we compare our work to related contributions, in particular to the framework of learning TBoxes from entailment queries, by Konev et al. [7,8]. We conclude in Section 7 with an overview of interesting open problems.

2 Description Logic Preliminaries

The language of the Description Logic (DL) $\mathcal{EL}$ [4] is given by (1) a vocabulary $\Sigma = (N_C, N_R)$, where N_C is a set of concept names (i.e., unary predicates, e.g., Father, Woman) and N_R a set of role names (i.e., binary predicates, e.g., $\mathsf{hasChild}$, likes), and (2) the following set of constructors for defining complex concepts, which shall be divided into two groups:

$$\begin{array}{ll} \mathcal{EL}: & C, D ::= \top \mid A \mid C \sqcap D \mid \exists r.C \\ \mathcal{L}^{\sqcap}: & C, D ::= \top \mid A \mid C \sqcap D \end{array}$$

where $A \in N_C$ and $r \in N_R$. Concept $\top$ denotes all individuals in the domain, $C \sqcap D$ the class of individuals that are instances of both C and D, and $\exists r.C$ describes all individuals that are related to some instance of C via the role r. The

[4] See `http://www.w3.org/TR/owl2-profiles/`.

set of $\mathcal{L}^{\sqcap}$ concepts naturally captures the propositional part of $\mathcal{EL}$. The *depth* of a subconcept D in C is the number of existential restrictions within the scope of which D remains. The *depth of a concept* C is the depth of its subconcept with the greatest depth in C. Every $\mathcal{L}^{\sqcap}$ concept is trivially of depth 0.

A *concept inclusion* (or a *TBox axiom*) is an expression of the form $C \sqsubseteq D$, stating that all individuals of type C are D. We sometimes write $C \equiv D$ as an abbreviation for two inclusions: $C \sqsubseteq D$ and $D \sqsubseteq C$. For instance, axioms (i) and (ii) below state, respectively, that (i) the class of mothers consists of all and only those individuals who are women and have at least one child, (ii) while every individual of type $\mathsf{Father\_of\_boy}$ is a father and has at least one male child:

$$\begin{array}{rcl} \mathsf{Mother} \equiv \mathsf{Woman} \sqcap \exists \mathsf{hasChild}.\top & & (i) \\ \mathsf{Father\_of\_boy} \sqsubseteq \mathsf{Father} \sqcap \exists \mathsf{hasChild}.\mathsf{Man} & & (ii) \end{array}$$

A TBox (or *ontology*) is a finite set of such concept inclusions in a particular language fragment. The language fragments considered in this paper are classified according to the type of restrictions imposed on the syntax of concepts C and D in the concept inclusions $C \sqsubseteq D$ permitted in the TBoxes:

$\mathcal{EL}$:	C and D are both $\mathcal{EL}$ concepts;
$\mathcal{EL}^{\mathrm{rhs}}$:	C is an $\mathcal{L}^{\sqcap}$ concept and D an $\mathcal{EL}$ concept;
$\mathcal{EL}^{\mathrm{lhs}}$:	C is an $\mathcal{EL}$ concept and D an $\mathcal{L}^{\sqcap}$ concept;
$\mathcal{L}^{\sqcap}$:	C and D are both $\mathcal{L}^{\sqcap}$ concepts.

For instance, a TBox consisting of axioms (i) and (ii) above, belongs to language $\mathcal{EL}$, as it in fact contains some $\mathcal{EL}^{\mathrm{rhs}}$ axioms ($\mathsf{Mother} \sqsubseteq \mathsf{Woman} \sqcap \exists \mathsf{hasChild}.\top$ and (ii)) as well as one $\mathcal{EL}^{\mathrm{lhs}}$ axiom ($\mathsf{Woman} \sqcap \exists \mathsf{hasChild}.\top \sqsubseteq \mathsf{Mother}$).

The semantics of DL languages is defined through interpretations of the form $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty *domain of individuals* and $\cdot^{\mathcal{I}}$ is an *interpretation function* mapping each $A \in N_C$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each $r \in N_R$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function is inductively extended over complex expressions according to the fixed semantics of the constructors:

$$\begin{array}{rcl} \top^{\mathcal{I}} & = & \Delta^{\mathcal{I}} \\ (C \sqcap D)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} \mid x \in C^{\mathcal{I}} \cap D^{\mathcal{I}}\} \\ (\exists r.C)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} \mid \exists y : (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} \end{array}$$

An interpretation $\mathcal{I}$ *satisfies* a concept inclusion $C \sqsubseteq D$ ($\mathcal{I} \models C \sqsubseteq D$) *iff* $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Whenever $\mathcal{I}$ satisfies all axioms in a TBox $\mathcal{T}$ ($\mathcal{I} \models \mathcal{T}$), we say that $\mathcal{I}$ is a *model* of $\mathcal{T}$. Interpretations and models defined in this way are in fact usual Kripke structures, which can be naturally represented as labelled graphs, with nodes representing individuals in the domain, edges — roles, and labels — the interpretations of concept and role names, respectively. For instance, the three graphs in Figure 1 all represent possible models of the TBox consisting of axioms (i) and (ii) above:

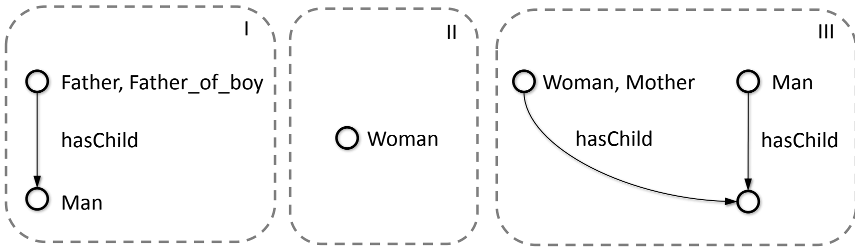

Fig. 1. Sample DL models.

Note that, as there are no a priori restrictions imposed on the number of domain individuals, a DL TBox might in general have infinitely many models of possibly infinite size. For a set of interpretations $\mathcal{S}$, we write $\mathcal{S} \models C \sqsubseteq D$ to denote that every interpretation in $\mathcal{S}$ satisfies $C \sqsubseteq D$. We say that $\mathcal{T}$ *entails* $C \sqsubseteq D$ ($\mathcal{T} \models C \sqsubseteq D$) *iff* every model of $\mathcal{T}$ satisfies $C \sqsubseteq D$. Two TBoxes $\mathcal{T}$ and $\mathcal{H}$ are (logically) *equivalent* ($\mathcal{T} \equiv \mathcal{H}$) *iff* they have the same sets of models.

A *pointed interpretation* $(\mathcal{I}, d)$ is a pair consisting of a DL interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ and an individual $d \in \Delta^{\mathcal{I}}$, such that every $e \in \Delta^{\mathcal{I}}$ different from d is reachable from d through some role composition in $\mathcal{I}$. By a slight abuse of notation, given an arbitrary DL interpretation $\mathcal{I}$ and an individual $d \in \Delta^{\mathcal{I}}$, we write $(\mathcal{I}, d)$ to denote the largest subset $\mathcal{I}'$ of $\mathcal{I}$ such that $(\mathcal{I}', d)$ is a pointed interpretation. If it is clear from the context, we refer to pointed interpretations and pointed models simply as interpretations and models. We say that $(\mathcal{I}, d)$ is a model of a concept C *iff* $d \in C^{\mathcal{I}}$; it is a model of C *w.r.t.* $\mathcal{T}$ whenever also $\mathcal{I} \models \mathcal{T}$.

An interpretation $(\mathcal{I}, d)$ can be *homomorphically embedded* in an interpretation $(\mathcal{J}, e)$, denoted as $(\mathcal{I}, d) \mapsto (\mathcal{J}, e)$, *iff* there exists a mapping $h : \Delta^{\mathcal{I}} \mapsto \Delta^{\mathcal{J}}$, satisfying the following conditions:

- $h(d) = e$,
- if $(a, b) \in r^{\mathcal{I}}$ then $(h(a), h(b)) \in r^{\mathcal{J}}$, for every $a, b \in \Delta^{\mathcal{I}}$ and $r \in N_R$,
- if $a \in A^{\mathcal{I}}$ then $h(a) \in A^{\mathcal{J}}$, for every $a \in \Delta^{\mathcal{I}}$ and $A \in N_C$.

A model $(\mathcal{I}, d)$ of C (w.r.t. $\mathcal{T}$) is called *minimal iff* it can be homomorphically embedded in every other model of C (w.r.t. $\mathcal{T}$). It is well-known that $\mathcal{EL}$ concepts and TBoxes always have such minimal models (unique up to homomorphic embeddings) [9]. As in most modal logics, arbitrary $\mathcal{EL}$ models can be unravelled into equivalent tree-shaped models. Finally, we observe that due to a tight relationship between the syntax and semantics of $\mathcal{EL}$, every tree-shaped interpretation $(\mathcal{I}, d)$ can be viewed as an $\mathcal{EL}$ concept $C_{\mathcal{I}}$, such that $(\mathcal{I}, d)$ is a minimal model of $C_{\mathcal{I}}$. Formally, we set $C_{\mathcal{I}} = C(d)$, where for every $e \in \Delta^{\mathcal{I}}$ we let $C(e) = \top \sqcap A(e) \sqcap \exists(e)$, with $A(e) = \bigsqcap\{A \in N_C \mid e \in A^{\mathcal{I}}\}$ and $\exists(e) = \bigsqcap_{(r,f) \in N_R \times \Delta^{\mathcal{I}} \text{ s.t. } (e,f) \in r^{\mathcal{I}}} \exists r.C(f)$. In that case we call $C_{\mathcal{I}}$ the *covering concept* for $(\mathcal{I}, d)$. For instance, the covering concept for model I in Figure 1 is $\top \sqcap \mathsf{Father} \sqcap \mathsf{Father\_of\_boy} \sqcap \exists \mathsf{hasChild}.(\top \sqcap \mathsf{Man})$, which can be simplified as $\mathsf{Father} \sqcap \mathsf{Father\_of\_boy} \sqcap \exists \mathsf{hasChild}.(\mathsf{Man})$.

3 Learning Model

The learning model studied in this paper is a variant of learning from positive interpretations [5,6]. In our setting, the teacher fixes a *target TBox* $\mathcal{T}$, whose set of all models is denoted by $\mathcal{M}(\mathcal{T})$. Further, the teacher presents a set of examples from $\mathcal{M}(\mathcal{T})$ to the learner, whose goal is to correctly identify $\mathcal{T}$ based on this input. The learning process is conducted relative to a mutually known DL language $\mathcal{L}$ and a finite vocabulary $\Sigma_\mathcal{T}$ used in $\mathcal{T}$.

In principle, $\mathcal{M}(\mathcal{T})$ contains sufficient information in order to enable correct identification of $\mathcal{T}$, as the following correspondence implies:

$$\mathcal{M}(\mathcal{T}) \models C \sqsubseteq D \quad \textit{iff} \quad \mathcal{T} \models C \sqsubseteq D, \text{ for every } C \sqsubseteq D \text{ in } \mathcal{L}.$$

However, as $\mathcal{M}(\mathcal{T})$ might consist of infinitely many models of possibly infinite size, the teacher cannot effectively present them all to the learner. Instead, the teacher must confine him- or herself to certain finitely presentable subset of $\mathcal{M}(\mathcal{T})$, called the *learning set*. For the sake of clarity, we focus here on the simplest case when learning sets consist of finitely many finite models.[5] Formally, we summarize the learning model with the following definitions.

Definition 1 (TIP). *A* TBox Identification Problem (TIP) *is a pair* $(\mathcal{T}, \mathcal{S})$*, where* $\mathcal{T}$ *is a* TBox *in a DL language* $\mathcal{L}$ *and* $\mathcal{S}$*, called the* learning set*, is a finite set of finite models of* $\mathcal{T}$*.*

Definition 2 (Learner, identification). *For a DL language* $\mathcal{L}$*, a* learner *is a computable function* G*, which for every set* $\mathcal{S}$ *over* $\Sigma_\mathcal{T}$ *returns a TBox in* $\mathcal{L}$ *over* $\Sigma_\mathcal{T}$*. Learner* G correctly identifies $\mathcal{T}$ *on* $\mathcal{S}$ *whenever* $G(\mathcal{S}) \equiv \mathcal{T}$*.*

Definition 3 (Learnability). *For a DL language* $\mathcal{L}$*, the class of TBoxes expressible in* $\mathcal{L}$ *is* learnable iff *there exists a learner* G *such that for every TBox* $\mathcal{T}$ *in* $\mathcal{L}$ *there exists a learning set* $\mathcal{S}$ *on which* G *correctly identifies* $\mathcal{T}$*. It is said to be* finitely learnable *whenever it is learnable from finite learning sets only.*

We are primarily interested here in the notion of finite learnability, as it provides a natural formal foundation for the task of ontology learning from data. By data, in the DL context, we understand collections of atomic concept and role assertions over domain individuals (e.g., $\mathsf{Father}(\mathsf{john})$, $\mathsf{hasChild}(\mathsf{john}, \mathsf{mary})$), which under certain assumptions regarding their structuring with respect to the background ontology can be seen as models of that ontology and, consequently, as potentially valuable learning sets. Figure 2 presents an example of a TIP with

[5] An alternative, more general approach can be defined in terms of specific fragments of models. Such generalization, which lies beyond the scope of this paper, is essential when the learning problem concerns languages without finite model property.

$$\mathsf{Mother} \equiv \mathsf{Woman} \sqcap \exists \mathsf{hasChild}.\top$$
$$\mathsf{Father} \equiv \mathsf{Man} \sqcap \exists \mathsf{hasChild}.\top$$
$$\mathsf{Father\_of\_boy} \equiv \mathsf{Father} \sqcap \exists \mathsf{hasChild}.\mathsf{Man}$$

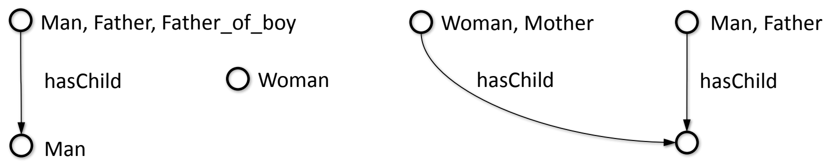

Fig. 2. A sample TIP with an $\mathcal{EL}$ TBox and a finite learning set

a finite learning set, which consists of a single model of the assumed ontology. The key question is then what formal criteria must this set satisfy to warrant correct identification of the ontology constraining it. To this end we employ the basic *admissibility condition*, characteristic also of other learning frameworks [10], which ensures that the learning set is sufficiently rich to enable precise discrimination between the correct hypothesis and all the incorrect ones.

Definition 4 (Admissibility). *A TIP* $(\mathcal{T}, \mathcal{S})$ *is* admissible iff *for every* $C \sqsubseteq D$ *in* $\mathcal{L}$ *such that* $\mathcal{T} \not\models C \sqsubseteq D$ *there exists* $\mathcal{I} \in \mathcal{S}$ *such that* $\mathcal{I} \not\models C \sqsubseteq D$.

For the target TBox $\mathcal{T}$, let $\mathcal{T}^{\not\models}$ be the set of all concept inclusions in $\mathcal{L}$ that are not entailed by $\mathcal{T}$, i.e., $\mathcal{T}^{\not\models} = \{C \sqsubseteq D \text{ in } \mathcal{L} \mid \mathcal{T} \not\models C \sqsubseteq D\}$. The admissibility condition requires that for every $C \sqsubseteq D \in \mathcal{T}^{\not\models}$, the learning set $\mathcal{S}$ must contain a "counterexample" for it, i.e., an individual $d \in \Delta^{\mathcal{I}}$, for some $\mathcal{I} \in \mathcal{S}$, such that $d \in C^{\mathcal{I}}$ and $d \notin D^{\mathcal{I}}$. Consequently, any learning set must contain such counterexamples to all elements of $\mathcal{T}^{\not\models}$, or else, the learner might never be justified to exclude some of these concept inclusions from the hypothesis. If it was possible to represent them finitely we could expect that ultimately the learner can observe all of them and correctly identify the TBox. In the next section, we investigate this prospect formally in different fragments of $\mathcal{EL}$.

4 Finite Learning Sets

As argued in the previous section, to enable finite learnability of $\mathcal{T}$ in a given language $\mathcal{L}$, the relevant counterexamples to all the concept inclusions not entailed by $\mathcal{T}$ must be presentable within a finite learning set $\mathcal{S}$. Firstly, we can immediately observe that this requirement is trivially satisfied for $\mathcal{L}^{\sqcap}$. Clearly, $\mathcal{L}^{\sqcap}$ can only induce finitely many different concept inclusions (up to logical equivalence) on finite vocabularies, such as $\Sigma_{\mathcal{T}}$. Hence, the set $\mathcal{T}^{\not\models}$ can always be finitely represented (up to logical equivalence) and it is straightforward to finitely present counterexamples to all its members. For more expressive fragments of $\mathcal{EL}$, however, this cannot be assumed in general, as the $\exists r.C$ constructor induces infinitely many concepts. One negative result comes with the case of $\mathcal{EL}$ itself, as demonstrated in the next theorem.

Theorem 1 (Finite learning sets in $\mathcal{EL}$). *Let $\mathcal{T}$ be a TBox in $\mathcal{EL}$. There exists no finite set $\mathcal{S}$ such that $(\mathcal{T}, \mathcal{S})$ is admissible.*

The full proof of this and subsequent results is included in an online technical report [11]. The argument rests on the following lemma. Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP and C a concept. By $\mathcal{S}(C)$ we denote the set of all models $(\mathcal{I}, d)$ of C w.r.t. $\mathcal{T}$ such that $\mathcal{I} \in \mathcal{S}$. By $\bigcap \mathcal{S}(C)$ we denote the intersection of all these models, i.e., the model $(\mathcal{J}, d)$, such that:

1. $(\mathcal{J}, d) \mapsto (\mathcal{I}, d)$ for every $(\mathcal{I}, d) \in \mathcal{S}(C)$,
2. for every other model $(\mathcal{J}', d)$ such that $(\mathcal{J}', d) \mapsto (\mathcal{I}, d)$ for every $(\mathcal{I}, d) \in \mathcal{S}(C)$ and $(\mathcal{J}, d) \mapsto (\mathcal{J}', d)$, it is the case that $(\mathcal{J}', d) \mapsto (\mathcal{J}, d)$.

Lemma 1 (Minimal model lemma). *Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP for $\mathcal{T}$ in $\mathcal{EL}$ (resp. in $\mathcal{EL}^{rhs}$), and C be an $\mathcal{EL}$ (resp. $\mathcal{L}^{\sqcap}$) concept. Whenever $\mathcal{S}(C)$ is non-empty then $\bigcap \mathcal{S}(C)$ is a minimal model of C w.r.t. $\mathcal{T}$.*

Given the lemma, we consider a concept inclusion of type:

$$\tau_n := \underbrace{\exists r. \ldots \exists r.}_{n} \top \sqsubseteq \underbrace{\exists r. \ldots \exists r. \exists r.}_{n+1} \top$$

Suppose $\tau_n \in \mathcal{T}^{\not\models}$ for some $n \in \mathbb{N}$. Since by the admissibility condition a counterexample to τ_n must be present in $\mathcal{S}$, it must be the case that $\mathcal{S}(C) \neq \emptyset$, where C is the left-hand-side concept in τ_n. By the lemma and the definition of a minimal model, it is easy to see that $\mathcal{S}$ must contain a finite chain of individuals of length exactly $n + 1$, as depicted below:

$$\underbrace{\bullet \xrightarrow{\;\;r\;\;} \bullet \ldots \bullet \xrightarrow{\;\;r\;\;} \bullet}_{n+1}$$

Finally, since there can always exist some $n \in \mathbb{N}$, such that $\tau_m \in \mathcal{T}^{\not\models}$ for every $m \geq n$, we see that the joint size of all necessary counterexamples in such cases must inevitably be also infinite. Consequently, for some $\mathcal{EL}$ TBoxes admissible TIPs based on finite learning sets might not exist, and so finite learnability cannot be achieved in general.

One trivial way to tame this behavior is to "finitize" $\mathcal{T}^{\not\models}$ by delimiting the entire space of possible TBox axioms to a pre-defined, finite set. This can be achieved, for instance, by restricting the permitted depth of complex concepts or generally setting some a priori bound on the size of axioms. Such ad hoc solutions, though likely efficient in practice, are not very elegant. As a more interesting alternative, we are able to show that there exist at least two languages between $\mathcal{L}^{\sqcap}$ and $\mathcal{EL}$, namely $\mathcal{EL}^{\text{lhs}}$ and $\mathcal{EL}^{\text{rhs}}$, for which finite learning sets are always guaranteed to exist, regardless of the fact that they permit infinitely many concept inclusions. In fact, we demonstrate that in both cases such learning sets might well consist of exactly one exemplary finite model.

We adopt the technique of so-called *types*, known from the area of modal logics [12]. Types are finite abstractions of possible individuals in the interpretation domain, out of which arbitrary models can be constructed. Let $con(\mathcal{T})$ be the set of all concepts (and all their subconcepts) occurring in $\mathcal{T}$. A *type* over $\mathcal{T}$ is a set $t \subseteq con(\mathcal{T})$, such that $C \sqcap D \in t$ *iff* $C \in t$ and $D \in t$, for every $C \sqcap D \in con(\mathcal{T})$. A type t is *saturated* for $\mathcal{T}$ *iff* for every $C \sqsubseteq D \in \mathcal{T}$, if $C \in t$ then $D \in t$. For any $S \subseteq con(\mathcal{T})$, we write t_S to denote the smallest saturated type containing S. It is easy to see, that t_S must be unique for $\mathcal{EL}$.

The next theorem addresses the case of $\mathcal{EL}^{\text{rhs}}$. Figure 3 illustrates a finite learning set for a sample $\mathcal{EL}^{\text{rhs}}$ TBox, following the construction in the proof.

Theorem 2 (Finite learning sets in $\mathcal{EL}^{\text{rhs}}$). *Let $\mathcal{T}$ be a TBox in $\mathcal{EL}^{\text{rhs}}$. There exists a finite set $\mathcal{S}$ such that $(\mathcal{T}, \mathcal{S})$ is admissible.*

Proof sketch. Let Θ be the smallest set of types satisfying the following conditions:

- $t_S \in \Theta$, for every $S \subseteq N_C$ and for $S = \{\top\}$,
- if $t \in \Theta$ then $t_{\{C\}} \in \Theta$, for every $\exists r.C \in t$.

We define the interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ as follows:

- $\Delta^{\mathcal{I}} := \Theta$,
- $t \in A^{\mathcal{I}}$ *iff* $A \in t$, for every $t \in \Theta$ and $A \in N_C$,
- $(t, t_{\{C\}}) \in r^{\mathcal{I}}$, for every $t \in \Theta$, whenever $\exists r.C \in t$.

Then $\mathcal{S} = \{\mathcal{I}\}$ is a finite learning set such that $(\mathcal{T}, \mathcal{S})$ is admissible. ❑

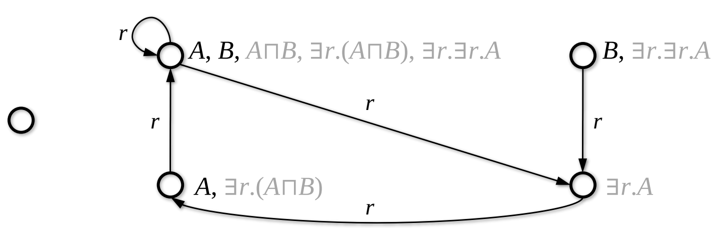

Fig. 3. A finite learning set for an $\mathcal{EL}^{\text{rhs}}$ TBox $\{A \sqsubseteq \exists r.(A \sqcap B), B \sqsubseteq \exists r.\exists r.A\}$. The figure includes type contents (in grey), as defined in the proof of Theorem 2.

A similar, though somewhat more complex construction demonstrates the existence of finite learning sets in $\mathcal{EL}^{\text{lhs}}$. Again, we illustrate the approach with an example in Figure 4.

Theorem 3 (Finite learning sets in $\mathcal{EL}^{\text{lhs}}$). *Let $\mathcal{T}$ be a TBox in $\mathcal{EL}^{\text{lhs}}$. There exists a finite set $\mathcal{S}$ such that $(\mathcal{T}, \mathcal{S})$ is admissible.*

Proof sketch. Let Θ be the set of all saturated types over $\mathcal{T}$, and Θ^* be its subset obtained by iteratively eliminating all those types t that violate the following condition: for every $r \in N_R$ and every existential restriction $\exists r.C \in t$ there is $u \in \Theta^*$ such that:

- $C \in u$,
- for every $\exists r.D \in con(\mathcal{T})$, if $D \in u$ then $\exists r.D \in t$.

Further, we define the interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ as follows:

- $\Delta^\mathcal{I} := \Theta^*$,
- $t \in A^\mathcal{I}$ *iff* $A \in S_t$, for every $t \in \Theta^*$ and $A \in N_C$,
- $(t, u) \in r^\mathcal{I}$ *iff* for every $\exists r.C \in con(\mathcal{T})$, if $C \in u$ then $\exists r.C \in t$.

Then $\mathcal{S} = \{\mathcal{I}\}$ is a finite learning set such that $(\mathcal{T}, \mathcal{S})$ is admissible. ❑

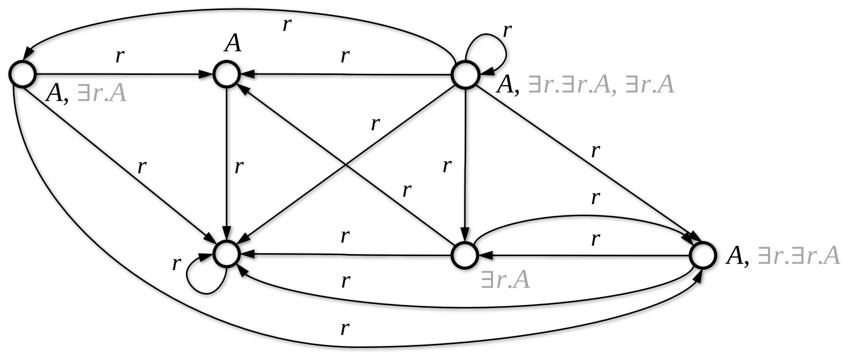

Fig. 4. A finite learning set for an $\mathcal{EL}^{\text{lhs}}$ TBox $\{\exists r.\exists r.A \sqsubseteq A\}$. The figure includes type contents (in grey), as defined in the proof of Theorem 3.

5 Learning Algorithms

In this section, we devise two learning algorithms for admissible TIPs with finite learning sets that correctly identify 1) $\mathcal{EL}^{\text{lhs}}$ and $\mathcal{EL}^{\text{rhs}}$ TBoxes using an equivalence oracle, and 2) $\mathcal{EL}^{\text{rhs}}$ TBoxes without such an oracle, i.e., in a fully unsupervised manner.

Since the set $\mathcal{T}^{\not\models} = \{C \sqsubseteq D \text{ in } \mathcal{L} \mid \mathcal{T} \not\models C \sqsubseteq D\}$ can be in general infinite, our starting observation is that a learner cannot effectively eliminate concept inclusions from $\mathcal{T}^{\not\models}$ using a straightforward enumeration, thus arriving at the target TBox $\mathcal{T}$. The only feasible strategy is to try to identify the "good" candidate axioms to be included in $\mathcal{T}$, and possibly apply the elimination strategy only to finitely many incorrect guesses. One generic procedure to employ such

Algorithm 1 Learning $\mathcal{EL}^{\text{rhs}}/\mathcal{EL}^{\text{lhs}}$ TBoxes on finite inputs.

Input: a TIP $(\mathcal{T}, \mathcal{S})$
Output: a hypothesis TBox $\mathcal{H}$

$n := 2$
$\mathcal{H}_n := \emptyset$
while '$\mathcal{H}_n \equiv \mathcal{T}$'? is 'NO' (*equivalence oracle querying*) **do**
  $n := n + 1$
  $\mathsf{Cand}_n := \{C \sqsubseteq D \in \mathcal{EL}^{\text{rhs}}/\mathcal{EL}^{\text{lhs}} \mid \ell(C \sqsubseteq D) = n\}$
  $\mathsf{Accept}_n := \{C \sqsubseteq D \in \mathsf{Cand}_n \mid \mathcal{S} \models C \sqsubseteq D\}$
  $\mathcal{H}_n := \mathcal{H}_{n-1} \cup \mathsf{Accept}_n$
end while
return $\mathcal{H}_n$

heuristic, which we define as Algorithm 1, attempts to construct the hypothesis by extending it with consecutive axioms of systematically growing size that are satisfied by the learning set. There, by $\ell(C \sqsubseteq D)$ we denote the size of the axiom $C \sqsubseteq D$ measured in the total number of symbols used for expressing this axiom. At each step the algorithm makes use of a simple equivalence oracle, which informs whether the currently considered hypothesis is already equivalent to the learning target (in that case the identification succeeds) or whether some axioms are still missing. Theorem 4 demonstrates the correctness of this approach.

Theorem 4 (Correct identification in $\mathcal{EL}^{\text{rhs}}/\mathcal{EL}^{\text{lhs}}$). *Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP for $\mathcal{T}$ in $\mathcal{EL}^{rhs}/\mathcal{EL}^{lhs}$. Then the hypothesis TBox $\mathcal{H}$ generated by Algorithm 1 is equivalent to $\mathcal{T}$.*

Obviously the use of the oracle is essential to warrant termination of the algorithm. It is not difficult to see that without it, the algorithm must still converge on the correct TBox for some $n \in \mathbb{N}$, and consequently settle on it, i.e., $\mathcal{H}_m \equiv \mathcal{H}_n$ for every $m \geq n$. However, at no point of time can it guarantee that the convergence has been already achieved, and so it can only warrant learnability in the limit. This result is therefore not entirely satisfactory considering we aim at finite learnability from data in the unsupervised setting.

A major positive result, on the contrary, can be delivered for the case of $\mathcal{EL}^{\text{rhs}}$, for which we devise an effective learning algorithm making no reference to any oracle. It turns out that in $\mathcal{EL}^{\text{rhs}}$ the "good" candidate axioms can be directly extracted from the learning set, thus granting a proper unsupervised learning method. The essential insight is provided by Lemma 1, presented in the previous section. Given any $\mathcal{L}^{\sqcap}$ concept C such that $\mathcal{S}(C) \neq \emptyset$ we are able to identify a tree-shaped minimal model of C w.r.t. $\mathcal{T}$. Effectively, it suffices to retrieve only the initial part of this model, discarding its infinitely recurrent (cyclic) subtrees. Such an initial model $\mathcal{I}_{\text{init}}$ is constructed by Algorithm 2. The algorithm performs simultaneous unravelling of all models in $\mathcal{S}(C)$, while on the way, computing intersections of visited combinations of individuals, which are subsequently added to the model under construction. Whenever the same

Algorithm 2 Computing the initial part of the minimal model $\bigcap \mathcal{S}(C)$

Input: the set $\mathcal{S}(C) = \{(\mathcal{I}_i, d_i)\}_{0 \leq i \leq n}$, for some $n \in \mathbb{N}$
Output: a finite tree-shaped interpretation $(\mathcal{J}, d)$, where $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$

1: $\Delta^{\mathcal{J}} := \{f(d_0, \ldots, d_n)\}$, for a "fresh" function symbol f
2: $A^{\mathcal{J}} := \emptyset$, for every $A \in N_C$
3: $r^{\mathcal{J}} := \emptyset$, for every $r \in N_R$
4: **for** every $f(d_0, \ldots, d_n) \in \Delta^{\mathcal{J}}$, $(e_0, \ldots, e_n) \in \Delta^{\mathcal{I}_0} \times \ldots \times \Delta^{\mathcal{I}_n}$, $r \in N_R$ **do**
5:     **if** $(d_i, e_i) \in r^{\mathcal{I}_i}$ for every $0 \leq i \leq n$ **and** there exists no function symbol g such that $g(e_0, \ldots, e_n)$ is an ancestor of $f(d_0, \ldots, d_n)$ in $\mathcal{J}$ **then**
6:         $\Delta^{\mathcal{J}} := \Delta^{\mathcal{J}} \cup \{g(e_0, \ldots, e_n)\}$, for a "fresh" function symbol g
7:         $r^{\mathcal{J}} := r^{\mathcal{J}} \cup \{(f(d_0, \ldots, d_n), g(e_0, \ldots, e_n))\}$
8:     **end if**
9: **end for**
10: **for** every $f(d_0, \ldots, d_n) \in \Delta^{\mathcal{J}}$, $A \in N_C$ **do**
11:     **if** $d_i \in A^{\mathcal{I}_i}$ for every $0 \leq i \leq n$ **then**
12:         $A^{\mathcal{J}} := A^{\mathcal{J}} \cup \{f(d_0, \ldots, d_n)\}$
13:     **end if**
14: **end for**
15: **return** $(\mathcal{J}, f(d_0, \ldots, d_n))$, where $f(d_0, \ldots, d_n)$ is the root of $\mathcal{J}$, created at step 1.

combination of individuals is about to be visited for the third time on the same branch it is skipped, as the cycle is evidently detected and further unravelling is unnecessary. The covering concept $C_{\mathcal{I}_{\text{init}}}$ for the resulting interpretation $\mathcal{I}_{\text{init}}$ is then included in the hypothesis within the axiom $C \sqsubseteq C_{\mathcal{I}_{\text{init}}}$. Meanwhile, all $\mathcal{L}^{\sqcap}$ concepts C such that $\mathcal{S}(C) = \emptyset$ are ensured to entail every $\mathcal{EL}$ concept, as implied by the admissibility condition. The contents of the hypothesis TBox are formally specified in Definition 5.

Definition 5 ($\mathcal{EL}^{\mathbf{rhs}}$ hypothesis TBox). *Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP for $\mathcal{T}$ in $\mathcal{EL}^{rhs}$ over the vocabulary $\Sigma_{\mathcal{T}}$. The* hypothesis TBox $\mathcal{H}$ *is the set consisting of all the following axioms:*

- *$C \sqsubseteq C_{\mathcal{I}_{init}}$ for every $\mathcal{L}^{\sqcap}$ concept C such that $\mathcal{S}(C) \neq \emptyset$, where $C_{\mathcal{I}_{init}}$ is the covering concept for the interpretation $(\mathcal{I}_{init}, d)$ generated by Algorithm 2 on $\mathcal{S}(C)$;*
- *$C \sqsubseteq \bigsqcap_{r \in N_R} \exists r. \bigsqcap N_C$ for every $\mathcal{L}^{\sqcap}$ concept C such that $\mathcal{S}(C) = \emptyset$.*

To better illustrate the learning procedure, we consider a simple TIP consisting of an $\mathcal{EL}^{\text{rhs}}$ TBox $\mathcal{T} = \{A \sqsubseteq \exists r.(A \sqcap B)\}$ and a finite learning set $\mathcal{S} = \{\mathcal{I}\}$, with $\mathcal{I}$ as depicted in Figure 5. The assumed vocabulary containing two concept names — A and B — induces four distinct $\mathcal{L}^{\sqcap}$ concepts, namely: $\top$, A, B and $A \sqcap B$. For every such concept C we identify the corresponding set of its all pointed models $\mathcal{S}(C)$ contained in the learning set. For instance, $\mathcal{S}(A) = \{(\mathcal{I}, e_2), (\mathcal{I}, e_3)\}$. Further, we use Algorithm 2 to compute the initial

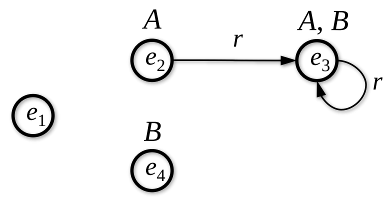

Fig. 5. A finite learning set for an $\mathcal{EL}^{\text{rhs}}$ TBox $\{A \sqsubseteq \exists r.(A \sqcap B)\}$.

part of the minimal model $\bigcap \mathcal{S}(C)$, as illustrated in Figure 6. Finally, based on these models we formulate the hypothesis TBox, as specified in Definition 5: $\mathcal{H} = \{\top \sqsubseteq \top, A \sqsubseteq \exists r.(A \sqcap B \sqcap \exists r.(A \sqcap B)), B \sqsubseteq B, A \sqcap B \sqsubseteq \exists r.(A \sqcap B)\}$. It is not difficult to verify that $\mathcal{H} \equiv \mathcal{T}$.

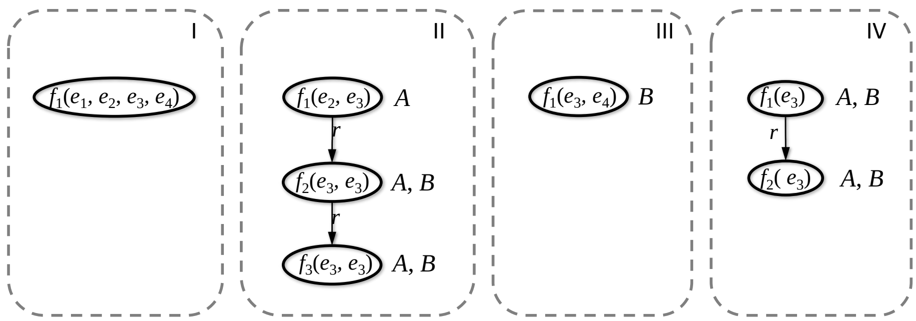

Fig. 6. The initial parts of the minimal models $\bigcap \mathcal{S}(C)$ computed with Algorithm 2 over the learning set in Figure 5, where (I) $C = \top$, (II) $C = A$, (III) $C = B$, (IV) $C = A \sqcap B$.

The correctness of the learning procedure is demonstrated in the following theorem.

Theorem 5 (Correct identification in $\mathcal{EL}^{\text{rhs}}$). *Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP for $\mathcal{T}$ in $\mathcal{EL}^{rhs}$. Then the hypothesis TBox $\mathcal{H}$ for $\mathcal{S}$ is equivalent to $\mathcal{T}$.*

Proof sketch. Let C be a concept in $\mathcal{L}^{\sqcap}$ such that $\mathcal{S}(C) \neq \emptyset$. By Lemma 1, the intersection $(\mathcal{J}, d) = \bigcap \mathcal{S}(C)$ is a minimal model of C w.r.t. $\mathcal{T}$. Without loss of generality, we assume that $(\mathcal{J}, d)$ is a tree-shaped model. We also note, that every $\mathcal{EL}$ concept C induces a syntactic tree, which corresponds directly to a minimal model of C. It is not difficult to see that Algorithm 2 indeed produces an initial part $(\mathcal{J}_{\text{init}}, d)$ of $(\mathcal{J}, d)$. By reconstructing the concept C_{init} from we in fact identify all minimal (i.e., necessary) consequences of C w.r.t. $\mathcal{T}$. However, certain infinite subtrees of $(\mathcal{J}, d)$ are omitted in $(\mathcal{J}_{\text{init}}, d)$. This happens due to the condition at step 5 of Algorithm 2, which terminates the construction of certain branches whenever a cycle is detected. In the rest of the proof, we show that the covering concept $C_{\mathcal{J}_{\mathrm{init}}}$ has the same minimal model w.r.t. $\mathcal{H}$ as C has w.r.t. $\mathcal{T}$. Since this is demonstrated to hold for every $\mathcal{L}^{\sqcap}$ concept C, we can conclude that $\mathcal{H} \equiv \mathcal{T}$. ❑

The learning algorithm runs in double exponential time in the worst case and generates TBoxes of double exponential size in the size of $\mathcal{S}$. This follows from the fact that the tree-shaped interpretations generated by Algorithm 2 might be of depth exponential in the number of individuals occurring in $\mathcal{S}$ and have exponential branching factor. Importantly, however, there might exist solutions far closer to being optimal which we have not as far investigated.

It is our strong conjecture, which we leave as an open problem, that a similar learning strategy should also be applicable in the context of $\mathcal{EL}^{\mathrm{lhs}}$.

6 Related Work

Ontology learning is an interdisciplinary research field drawing on techniques from Formal Concept Analysis [13,14], Natural Language Processing [15,16] and machine learning [6,17,16], to name a few. One classification of ontology learning techniques distinguishes between investigations of exact learnability, and approaches incorporating probabilistic, vague or fuzzy reasoning [18,19]. Another classification is at the level at which learning takes place [15] — does the problem address learning of concepts, concept hierarchies, logical theories or rules? Lehmann and Völker [17] distinguishes between four types of ontology learning: learning from text, data mining, concept learning and crowdsourcing.

In this landscape, the present paper is on exact learnability and, within this framework, addresses the problem of learning logical theories. That is, we address the problem at the level of relationships between concepts, positing a logical theory, rather than at the concept level, learning concept descriptions [20,21,22,23,24]. Furthermore, the target theory is identified from interpretations, and is hence related to various contributions on learnability of different types of formal structures from data, e.g.: first-order theories from facts [10], finite automata descriptions from observations [25], and logic programs from interpretations [5,6].

The model for exact learning of DL TBoxes which offers the most direct comparison to ours was introduced recently by Konev, et al. [8], and follows on prior research by the same authors based on Angluin's model of learning from entailment [7,26]. In their learning framework for learning from data retrieval queries, the learner identifies the TBox by posing two types of queries to an oracle: membership queries of the form "$(\mathcal{T}, \mathcal{A}) \models q$?", where $\mathcal{A}$ is a given ABox and q is a query, and equivalence queries of the form "Does the hypothesis ontology $\mathcal{H}$ entail the target ontology $\mathcal{T}$?". The authors study polynomial learnability in fragments of $\mathcal{EL}$ and DL-Lite, and for queries ranging from atomic to conjunctive queries.

Essentially, given a finite learning set in an admissible TIP, a learner from interpretations can autonomously answer arbitrary membership queries, thus effectively simulating the membership oracle. However, it does not have by default access to an equivalence oracle. Once such an oracle is included, as in Algorithm 1, the learning power of both learners becomes comparable for the languages investigated in the present paper. In this sense, our Theorem 4 should be also indirectly derivable from the results by Konev et al. However, our stronger result for $\mathcal{EL}^{\mathrm{rhs}}$ in Theorem 5 demonstrates that, at least in some cases, the learner from interpretations is able to succeed without employing any oracle. While learning from ABoxes and query answers makes sense in a semi-automated learning environment, learning from interpretations is in our view a more appropriate model in the context of fully autonomous learning.

7 Conclusions and Outlook

In this paper, we have delivered initial results on finite learnability of DL TBoxes from interpretations. We believe that this direction shows promise in establishing formal foundations for the task of ontology learning from data. Some immediate problems that are left open with this work concern finite learnability of $\mathcal{EL}^{\mathrm{lhs}}$ TBoxes in an unsupervised setting, and possibly of other lightweight fragments of DLs. Another set of very interesting research questions should deal, in our view, with the possibility of formulating alternative conditions on the learning sets and the corresponding learnability guarantees they would imply in different DL languages. In particular, some limited use of closed-world operator over the learning sets might allow to relax the practically restrictive admissibility condition. Finally, the development of practical learning algorithms, possibly building on existing inductive logic programming methods, is an obvious area to welcome further research efforts.

References

1. Maedche, A., Staab, S.: Ontology learning. In Staab, S., Studer, R., eds.: Handbook on Ontologies, Springer (2004) 173–189
2. Hoekstra, R.: The knowledge reengineering bottleneck. Journal of Semantic Web **1**(1,2) (2010) 111–115
3. Baader, F., Calvanese, D., Mcguinness, D.L., Nardi, D., Patel-Schneider, P.F.: The description logic handbook: theory, implementation, and applications. Cambridge University Press (2003)
4. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proc. of IJCAI-05. (2005)
5. De Raedt, L., Lavrač, N.: The many faces of inductive logic programming. In: Methodologies for Intelligent Systems. (1993) 435–449
6. De Raedt, L.: First order jk-clausal theories are PAC-learnable. Artificial Intelligence **70** (1994) 375–392
7. Konev, B., Lutz, C., Ozaki, A., Wolter, F.: Exact learning of lightweight description logic ontologies. In: Proc. of Principles of Knowledge Representation and Reasoning (KR-14). (2014)

8. Konev, B., Lutz, C., Wolter, F.: Exact learning of TBoxes in $\mathcal{EL}$ and DL-Lite. In: Proc. of the 28th International Workshop on Description Logics. (2015)
9. Lutz, C., Piro, R., Wolter, F.: Enriching $\mathcal{EL}$-concepts with greatest fixpoints. In: Proc. of the 19th European Conference on Artificial Intelligence (ECAI 2010), IOS Press (2010) 41–46
10. Shapiro, E.Y.: Inductive inference of theories from facts. In: Computational Logic: Essays in Honor of Alan Robinson (1991). MIT Press (1981)
11. Klarman, S., Britz, K.: Ontology learning from interpretations in lightweight description logics. Technical report, CSIR Centre for Artificial Intelligence Research, South Africa `http://klarman.synthasite.com/resources/KlaBri-ILP15.pdf` (2015)
12. Pratt, V.: Models of program logics. In: Proc. of Foundations of Computer Science (FOCS-79). (1979)
13. Baader, F., Ganter, B., Sertkaya, B., Sattler, U.: Completing description logic knowledge bases using formal concept analysis. In: Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07). (2007)
14. Distel, F.: Learning Description Logic Knowledge Bases from Data using Methods from Formal Concept Analysis. PhD thesis, TU Dresden (2011)
15. Buitelaar, P., Cimeano, P., Magnini, F., eds.: Ontology Learning from Text: Methods, Evaluation and Applications. IOS Press (2005)
16. Cimeano, P., Mädche, A., Staab, S., Völker, J.: Ontology learning. In: Handbook on Ontologies. Springer-Verlag (2009)
17. Lehmann, J., Völker, J., eds.: Perspectives on Ontology Learning. IOS Press (2014)
18. Cohen, W., Hirsh, H.: The learnability of description logics with equality constraints. Machine Learning **17**(2-3) (1994) 169–199
19. Lisi, F., Straccia, U.: A FOIL-like method for learning under incompleteness and vagueness. In: Proc. of the 23rd International Conference on Inductive Logic Programming, Springer-Verlag (2013) 123–139
20. Badea, L., Nienhuys-Cheng, S.H.: A refinement operator for description Logics. In Cussens, J., Frisch, A., eds.: Proc. of the 10th International Conference on Inductive Logic Programming. Volume 1866 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2000) 40–59
21. Lehmann, J., Hitzler, P.: A refinement operator based learning algorithm for the $\mathcal{ALC}$ description logic. In Blockeel, H., Ramon, J., Shavlik, J., Tadepalli, P., eds.: Inductive Logic Programming. Springer Berlin Heidelberg (2008) 147–160
22. Fanizzi, N., dAmato, C., Esposito, F.: DL-FOIL concept learning in description logics. Inductive Logic Programming (2008) 107–121
23. Cohen, W.W., Hirsh, H.: Learning the classic description logic: Theoretical and experimental results. In: Proc. of Principles of Knowledge Representation and Reasoning (KR-94). (1994)
24. Chitsaz, M., Wang, K., Blumenstein, M., Qi, G.: Concept learning for $\mathcal{EL}++$ by refinement and reinforcement. In: Proc. of the 12th Pacific Rim International Conference on Artificial Intelligence (PRICAI-12). (2012) 15–26
25. Pitt, L.: Inductive inference, DFAs, and computational complexity. In Jantke, K.P., ed.: Analogical and Inductive Inference. Volume 397 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (1989) 18–44
26. Angluin, D.: Queries and concept learning. Machine Learning **2**(4) (1988) 319–342

Appendix

Proof of Lemma 1.

Suppose on the contrary that $\bigcap \mathcal{S}(C) = (\mathcal{J}, d)$ is not a minimal model of C w.r.t. $\mathcal{T}$, but instead some other $(\mathcal{I}, d)$ is. In such case $(\mathcal{I}, d) \mapsto (\mathcal{J}, d)$, but the inverse does not hold. Let further D be a concept such that $d \in D^{\mathcal{J}}$ but $d \notin D^{\mathcal{I}}$. Clearly, such concept can always be created by ensuring that D expresses (part of) the difference between $(\mathcal{J}, d)$ and $(\mathcal{I}, d)$. By definition of $\bigcap \mathcal{S}(C)$, it follows that $d \in D^{\mathcal{I}_i}$, for every $\mathcal{I}_i \in \mathcal{S}$ and $d \in C^{\mathcal{I}_i}$. Consequently, we observe that $\mathcal{T} \not\models C \sqsubseteq D$, but a counterexample to $C \sqsubseteq D$ does not exist in $\mathcal{S}$. But this would imply that $(\mathcal{T}, \mathcal{S})$ is not admissible, which contradicts the assumption. ❑

Proof of Theorem 1.

Consider a trivial TBox $\{\exists r.\top \sqsubseteq \exists r.\top\}$ and a concept template $C_n = \exists r. \ldots \exists r.\top$, where for $n \in \mathbb{N}$ the depth of C_n equals n. It is easy to verify that $\mathcal{S}(C) \neq \emptyset$ (since, e.g., $\mathcal{T} \not\models C_n \sqsubseteq C_{n+1}$). Then, by Lemma 1, $\bigcap \mathcal{S}(C_n) = (\mathcal{I}, d)$ must be a minimal pointed model of C_n w.r.t. $\mathcal{T}$, where $\mathcal{T}$ is in fact negligible due to its trivial form. It is easy to observe that $(\mathcal{I}, d)$ must consist of exactly $n + 1$ distinct individuals connected in a single acyclic path of r-successors. Since there are infinitely many distinct concepts of the form C_n, it follows by induction that $\mathcal{S}$ must contain infinitely many distinct pointed models. ❑

Proof of Theorem 2.

Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation as defined in the proof sketch of Theorem 2 in Section 4. We claim that:

1. $\mathcal{I} \models \mathcal{T}$.
2. $\mathcal{I} \not\models C \sqsubseteq D$, for every $C \sqsubseteq D \in \mathcal{T}^{\not\models}$.

Claim 1. Follows immediately by construction of $\mathcal{I}$. For an arbitrary axiom $C \sqsubseteq D \in \mathcal{T}$ and an individual $t \in \Delta^{\mathcal{I}}$, we show that whenever $t \in C^{\mathcal{I}}$ then $t \in D^{\mathcal{I}}$. Suppose that indeed $t \in C^{\mathcal{I}}$. Since $C = \bigsqcap_i A_i$, for $A_i \in N_C$, therefore $t \in A_i^{\mathcal{I}}$, for every A_i, and so $A_i \in t$. Consequently, $C \in t$ follows by the definition of a type. But then it must be the case that $D \in t$, as Θ consists only of saturated types. Further, every conjunct of D must be in t, by the definition of type. Among these, for every atom $A \in t$, we have that $t \in A^{\mathcal{I}}$, while for every existential restriction $\exists r.B \in t$, there must exist another type $t_{\{B\}}$, such that $(t, t_{\{B\}}) \in r^{\mathcal{I}}$ and $B \in t_{\{B\}}$. By structural induction over B and the construction of $\mathcal{I}$ it follows that indeed $t \in D^{\mathcal{I}}$.

Claim 2. First we observe that, for arbitrary concepts C, D, if $t_{\{C\}} \in D^{\mathcal{I}}$, then it must be the case that $\mathcal{T} \models C \sqsubseteq D$. This follows from two basic observations. Firstly, $t_{\{C\}}$ contains only those concepts $A \in N_C$ such that $\mathcal{T} \models C \sqsubseteq A$. Secondly, $\mathcal{I}$ is constructed as a minimal model, in the sense that it includes only necessary successors, i.e., those entailed by respective existential restrictions, with minimal interpretations necessary to satisfy these restrictions. Further, we note that by construction of $\mathcal{I}$, for an arbitrary axiom $C \sqsubseteq D \in \mathcal{T}^{\not\models}$ there exists an individual $t \in \Delta^{\mathcal{I}}$, such that $t \in C^{\mathcal{I}}$. Clearly, since $C = \bigsqcap_i A_i$ or $C = \top$, therefore $t_{\{C\}} \in \Theta$ by the construction of Θ. We can therefore conclude that if $t_{\{C\}} \in D^{\mathcal{I}}$ then it would have to be the case that $\mathcal{T} \models C \sqsubseteq D$, which contradicts the assumption. ❑

Proof of Theorem 3.

Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation as defined in the proof sketch of Theorem 3 in Section 4. We claim that:

1. $\mathcal{I} \models \mathcal{T}$.
2. $\mathcal{I} \not\models C \sqsubseteq D$, for every $C \sqsubseteq D \in \mathcal{T}^{\not\models}$.

Claim 1. For an arbitrary $C \sqsubseteq D \in \mathcal{T}$ and $t \in \Theta^*$, suppose that $t \in C^{\mathcal{I}}$. Let $\mathcal{J}_C = (\mathcal{J}, w)$ be a minimal tree-shaped model of C. Clearly, $(\mathcal{J}, w) \mapsto (\mathcal{I}, t)$ via some mapping h. Consider any leaf individual d in $\mathcal{J}_C$, of depth greater than 0, and any concept $A \in N_C$, such that $d \in A^{\mathcal{J}_C}$. Clearly, the existence of this individual and its given interpretation must be justifiable by the fact that some subconcept $\exists r.B$ of C is satisfied in an r-predecessor of d in $\mathcal{J}_C$. But then, by construction of Θ^* and $\mathcal{I}$, it must be the case that $\exists r.B \in h(d)$, or else $h(d)$ would have been eliminated from Θ. By the bottom-up structural induction over the tree $\mathcal{J}_C$ and the syntax of C it follows that $C \in t$. But since t is saturated, therefore also $D \in t$. Consequently, by definition of types and construction of $\mathcal{I}$, since $D = \bigsqcap_i A_i$ (or $D = \top$), then for every conjunct A_i, it must be the case that $t \in A_i^{\mathcal{I}}$ ($t \in \top^{\mathcal{I}}$ follows trivially).

Claim 2. For an arbitrary axiom $C \sqsubseteq D \in \mathcal{T}^{\not\models}$, we show that there exists $d \in \Delta^{\mathcal{I}}$, such that $d \in C^{\mathcal{I}}$ and $d \notin D^{\mathcal{I}}$. Let $\mathcal{J}_C^{\mathcal{T}} = (\mathcal{J}, w)$ be a minimal model of C w.r.t. $\mathcal{T}$. It is easy to see that $w \in D^{\mathcal{J}_C^{\mathcal{T}}}$ *iff* $\mathcal{T} \models C \sqsubseteq D$, and therefore $w \notin D^{\mathcal{J}_C^{\mathcal{T}}}$. Further, we fix a mapping h from individuals in $\mathcal{J}_C^{\mathcal{T}}$ to corresponding types, such that $h(d) = \{A \in N_C \mid d \in A^{\mathcal{J}_C^{\mathcal{T}}}\} \cup \{C \sqcap D \in con(\mathcal{T}) \mid C, D \in h(d)\} \cup \{\exists r.C \in con(\mathcal{T}) \mid C \in h(e),\ (d, e) \in r^{\mathcal{J}_C^{\mathcal{T}}}\}$. It is not difficult to see that all such types are saturated and belong to Θ^*. But this means that by the construction of $\mathcal{I}$, there must exist an individual $d \in \Delta^{\mathcal{I}}$, such that $(\mathcal{J}_C^{\mathcal{T}}, w) \mapsto (\mathcal{I}, d)$, and so that $d \in C^{\mathcal{I}}$ and $d \notin D^{\mathcal{I}}$. ❑

Proof of Theorem 4.

Follows immediately. Clearly, whenever the oracle responds 'YES' the hypothesis TBox $\mathcal{H}$ is equivalent to $\mathcal{T}$. It therefore suffices to show that the algorithm always terminates in finite time. Let k be the size of the largest axiom in $\mathcal{T}$. Whenever the counter n reaches $k+1$, the algorithm must have already listed and verified all the (finitely many) axioms of the size smaller or equal to $k+1$ — in particular, all the axioms in $\mathcal{T}$. As the algorithm accepts only axioms that are satisfied in $\mathcal{S}$, i.e., entailed by $\mathcal{T}$, it means that at this stage the oracle must accept the TBox as it contains all and only the axioms from $\mathcal{T}$ and possibly their logical consequences. ❑

Proof of Theorem 5.

We start by noting that every $\mathcal{EL}$ concept C induces a syntactic tree, which corresponds to a minimal model of C. Observe, that C can always be equivalently represented as a concept $C = \top \sqcap \bigsqcap_i A_i \sqcap \bigsqcap_j \exists r_j.C_j$, where for every $i, j \geq 0$, it holds that $A_i \in N_C$, $r_j \in N_R$ and C_j is a possibly complex $\mathcal{EL}$ concept. We say that C *maps* to a node $w \in W$ of a labeled tree $(W, <, l)$ *iff* the following conditions are satisfied:

- $l(w) = \{A_i\}_i$
- for every j there exists $w_j \in W$ such that:
  - $(w, w_j) \in <$,
  - $l(w, w_j) = r_j$,
  - C_j maps to w_j.

The *syntactic tree* of an $\mathcal{EL}$ concept C is the smallest tree $C_T = (W, <, l)$ such that C maps to its root. By the semantics of $\mathcal{EL}$, it follows that C_T can be essentially viewed as a tree-shaped model $\mathcal{I}_C = (\mathcal{I}, w)$ of C, as it is possible to straightforwardly turn it into one as follows:

- $\Delta^{\mathcal{I}} = W$, where $w \in W$ is the root of C_T
- $w \in A^{\mathcal{I}}$ *iff* $A \in l(w)$, for every $w \in \Delta^{\mathcal{I}}$ and $A \in N_C$,
- $(w, v) \in r^{\mathcal{I}}$ *iff* $(w, v) \in <$ and $l(w, v) = r$, for every $w, v \in \Delta^{\mathcal{I}}$ and $r \in N_R$.

In fact, it is easy to see that $\mathcal{I}_C$ is a minimal model of C.

Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP with $\mathcal{T}$ in $\mathcal{EL}^{\mathrm{rhs}}$. Let C be a concept in $\mathcal{L}^{\sqcap}$ such that $\mathcal{S}(C) \neq \emptyset$. By Lemma 1, the intersection $(\mathcal{J}, d) = \bigcap \mathcal{S}(C)$ is a minimal model of C w.r.t. $\mathcal{T}$. Without loss of generality, we assume that $(\mathcal{J}, d)$ is a tree-shaped model. It is not difficult to see that Algorithm 2 indeed produces an initial part $(\mathcal{J}_{\mathrm{init}}, d)$ of $(\mathcal{J}, d)$. However, certain infinite subtrees of $(\mathcal{J}, d)$ are omitted in $(\mathcal{J}_{\mathrm{init}}, d)$. This happens due to the condition at step 5 of Algorithm 2, which terminates the construction of certain branches whenever a cycle is detected. In the following, we show that the covering concept $C_{\mathcal{J}_{\mathrm{init}}}$ has the same minimal model w.r.t. $\mathcal{H}$ as C has w.r.t. $\mathcal{T}$. Since, as will be demonstrated, this holds for every $\mathcal{L}^{\sqcap}$ concept C, we can conclude that $\mathcal{H} \equiv \mathcal{T}$.

First, we split the TBox $\mathcal{T}$ into the propositional and the relational components: $\mathcal{T}_P$ and $\mathcal{T}_R$, respectively. The set $\mathcal{T}_P$ consists of axioms $C \sqsubseteq C_P$, where $C \in \mathcal{L}^{\sqcap}$ and $C_P = \bigsqcap S$ for $S \subseteq N_C$ being the set of all atoms $A \in N_C$, such that $\mathcal{T} \models C \sqsubseteq A$. Whenever $S = \emptyset$ we set $C_P = \top$. For a $\mathcal{L}^{\sqcap}$ concept C let

$C_\exists$ be the set of all subconcepts of the form $\exists r.E_j$ occurring at depth 0 in some axiom $D \sqsubseteq E \in \mathcal{T}$ such that $\mathcal{T}_P \models C \sqsubseteq D$. We define $\mathcal{T}_R$ as the set of all axioms $C \sqsubseteq C_R$, for every $C \in \mathcal{L}^\sqcap$, where $C_R = \bigsqcap C_\exists$. Whenever $C_\exists = \emptyset$ we set $C_R = \top$. Clearly, $\mathcal{T}_P$ and $\mathcal{T}_R$ can be simply defined by looking up through the structure of the axioms' left-hand-side and right-hand-side. It is also easy to see that $\mathcal{T} \equiv \mathcal{T}_P \cup \mathcal{T}_R$. Finally, we observe that for every $C \in \mathcal{L}^\sqcap$ there exist unique axioms $C \sqsubseteq C_P \in \mathcal{T}_P$ and $C \sqsubseteq C_R \in \mathcal{T}_R$. We say that C_P and C_R are the $\mathcal{T}_P$- and $\mathcal{T}_R$-*consequents of* C, respectively.

For a $\mathcal{L}^\sqcap$ concept C, the $\mathcal{T}$-*saturation* of C is a concept $C^\mathcal{T}$ obtained by iteratively replacing every (maximal) $\mathcal{L}^\sqcap$ subconcept C' of C with the concept $D \sqcap E$, where D is the $\mathcal{T}_P$-consequent of C' and E is the $\mathcal{T}_R$-consequent of D. Naturally, $C^\mathcal{T}$ can be effectively an infinite concept. By the construction of $C^\mathcal{T}$, and properties of $\mathcal{T}_P$ and $\mathcal{T}_R$, it follows immediately that the interpretation $\mathcal{I}_{C^\mathcal{T}}$ corresponding to the syntactic tree of $C^\mathcal{T}$ is the minimal tree-shaped model of C w.r.t. $\mathcal{T}$. This also holds for any concept $C^{\mathcal{T}'}$ obtained from $C^\mathcal{T}$ by replacing its any subconcept $\exists r.C_1 \sqcap \exists r.C_2$ with $\exists r.C_1$ whenever $\emptyset \models C_1 \sqsubseteq C_2$. Hence there must exist some $C^{\mathcal{T}'}$ such that $(\mathcal{I}_{C^{\mathcal{T}'}}, d) = (\mathcal{J}, d)$. Without loss of generality, we assume that it is in fact $(\mathcal{I}_{C^\mathcal{T}}, d) = (\mathcal{J}, d)$ and consider further the concept $C^\mathcal{T}$ as the covering concept for $(\mathcal{J}, d)$.

Consider any node t in the syntactic tree $C^\mathcal{T}_T$. By D_t we denote the concept corresponding to the subtree of $C^\mathcal{T}_T$ rooted at t. Clearly, $D_t = \top \sqcap \bigsqcap_i A_i \sqcap \bigsqcap_j \exists r_j.C_j$, for some atoms $A_i \in N_C$, roles $r_j \in N_R$ and (possibly) complex concepts C_j. Let t' be the r-predecessor of t in $C^\mathcal{T}_T$, for some $r \in N_R$. Then $D_{t'}$ must contain $\exists r.D_t$ as one of its subconcepts at the depth 0. Consider further some cycle occurring in $(\mathcal{J}, d)$, i.e., let e and e' be two individuals in $(\mathcal{J}, d)$ such that $(\mathcal{J}, e) = (\mathcal{J}, e')$ and e' is a (possibly indirect) successor of e. In that case there must exist the corresponding nodes t, t' in $C^\mathcal{T}_T$, such that $D_t = D_{t'}$ and t' is a (possibly indirect) successor of t. Let $\tau = t_1, \ldots, t_n$, for some $n \geq 1$ be the path connecting t with t' in $C^\mathcal{T}_T$, where $t_1 = t$ and $t_n = t'$, and $l(t_i, t_{i+1}) = r_i$ for some $r_i \in N_R$, for every $1 \leq i \leq n-1$. We can observe that $D_{t_{n-1}}$ must contain the concept $\exists r_{n-1}.D_{t'}$ at the depth 0. Observe that $C_{\mathcal{J}_{\text{init}}}$ covers the part of $\mathcal{J}$ containing all nodes from τ except for t_n (and analogically on other branches at which recurrent subtrees are rooted). However, since concept $D_{t_{n-1}}$ can be clearly derived at the depth of t_{n-1} while constructing the $\mathcal{T}$-saturation of C, it follows that the $\mathcal{T}$-saturation of $C_{\mathcal{J}_{\text{init}}}$ must be equal to $C_\mathcal{T}$ w.r.t. $\mathcal{T}$, modulo removal of some possibly duplicate branches. Now we need to show, that this will be also the case when $\mathcal{T}$ is replaced with $\mathcal{H}$. This follows naturally by induction. For $C_{\mathcal{J}_{\text{init}}} = \top \sqcap \bigsqcap_i A_i \sqcap \bigsqcap_j \exists r_j.C_j$ we fix $\mathcal{T}'_R = (\mathcal{T}_R \setminus \{C \sqsubseteq E\}) \cup \{C \sqsubseteq \bigsqcap_j \exists r_j.C_j\}$. Note that the propositional consequent of C remains unchanged in $\mathcal{H}$ since $C_{\mathcal{J}_{\text{init}}}$ is always of depth at least one. Then the construction of the $(\mathcal{T}_P \cup \mathcal{T}'_R)$-saturation of C will progress via exactly the same steps as the $\mathcal{T}$-saturation of C, modulo application of the axiom $C \sqsubseteq \bigsqcap_j \exists r_j.C_j$ instead of $C \sqsubseteq E$. Effectively, at every occurrence of the concept C in the construction of the $(\mathcal{T}_P \cup \mathcal{T}'_R)$-saturation of C we attach the syntactic tree of $C_{\mathcal{J}_{\text{init}}}$, which is exactly as $C_\mathcal{T}$ down to the occurrence of cyclic subtrees. But then by repeating the inductive argument, we observe that ultimately this construction must lead to a tree equal to $C_{\mathcal{T}}$. Since the minimal model of C remains the same w.r.t. $(\mathcal{T}_P \cup \mathcal{T}'_R)$, we can proceed analogically with every $\mathcal{L}^{\sqcap}$ concept C and its corresponding axiom $C \sqsubseteq E \in \mathcal{T}'_R$, thus iteratively replacing the content of $\mathcal{T}_P \cup \mathcal{T}_R$ with $\mathcal{H}$.

We can therefore conclude that for every $\mathcal{L}^{\sqcap}$ concept C, such that $\mathcal{S}(C) \neq \emptyset$, we always obtain the same minimal model of C w.r.t. $\mathcal{H}$ as w.r.t. $\mathcal{T}$, provided the same property holds whenever $\mathcal{S}(C) = \emptyset$. Let us then finally consider a case of $\mathcal{L}^{\sqcap}$ concept C such that $\mathcal{S}(C) = \emptyset$. By admissibility of $\mathcal{S}$ this implies that $C \sqsubseteq D$, for every $\mathcal{EL}$ concept D expressible in the given vocabulary. In such case our learning algorithm includes the axiom $C \sqsubseteq \bigsqcap_{r \in N_R} \exists r. \bigsqcap N_C$ in $\mathcal{H}$. We can observe, however, that in this situation it must be also the case that $\mathcal{T} \models \bigsqcap N_C \sqsubseteq \bigsqcap_{r \in N_R} \exists r. \bigsqcap N_C$. Suppose the latter is not the case. This would mean that $\mathcal{S}(\bigsqcap N_C) \neq \emptyset$. But this would also imply that $\mathcal{S}(C) \neq \emptyset$, which contradicts the assumption. ❑