Towards Unsupervised Ontology Learning from Data

Szymon Klarman
Department of Computer Science
Brunel University London

Katarina Britz
Centre for AI Research
CSIR and 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 [Maedche and Staab, 2004]. 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 [Hoekstra, 2010]. 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 gap is addressed in the present work.

In this paper, 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 [Baader *et al.*, 2003], which have risen to prominence as, among others, the logics underpinning different profiles of the Web Ontology Language OWL[1]. In this paper, we focus on the lightweight DL $\mathcal{EL}$ [Baader *et al.*, 2005] 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 [De Raedt and Lavrač, 1993; De Raedt, 1994]. 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 $\mathcal{EL}^{\text{rhs}}$ and $\mathcal{EL}^{\text{lhs}}$, 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 $\mathcal{EL}^{\text{rhs}}$ and $\mathcal{EL}^{\text{lhs}}$ TBoxes from finite data, employing a basic equivalence oracle. Further, in case of $\mathcal{EL}^{\text{rhs}}$, 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. [Konev *et al.*, 2014]. We conclude in Section 7 with an overview of interesting open problems.

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

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

2 Description Logic Preliminaries

The language of the Description Logic (DL) $\mathcal{EL}$ [Baader *et al.*, 2005] is given by (1) a vocabulary $\Sigma = (N_C, N_R)$, where N_C is a set of concept names and N_R a set of role names, 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 $r \in N_R$. The 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.

The semantics 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:

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

A *concept inclusion* is an expression of the form $C \sqsubseteq D$, stating that all individuals of type C are D, as in, e.g.: $\mathsf{Father\_of\_son} \sqsubseteq \mathsf{Man} \sqcap \exists \mathsf{hasChild}.\mathsf{Man}$. The language fragments considered in this paper are categorized w.r.t. restrictions imposed on the syntax of concepts C and D in permitted concept inclusions $C \sqsubseteq D$:

$\mathcal{EL}$:	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.

A TBox (or *ontology*) is a finite set of concept inclusions, also called the *TBox axioms*, in a given language fragment.

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 $\mathcal{T}$. 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) [Lutz *et al.*, 2010]. 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 $(\mathcal{I}, d)$.

3 Learning Model

The learning model studied in this paper is a variant of learning from positive interpretations [De Raedt and Lavrač, 1993; De Raedt, 1994]. 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 signature $\Sigma_{\mathcal{T}}$ used in $\mathcal{T}$. Obviously, $\mathcal{M}(\mathcal{T})$ contains in principle sufficient information in order to enable correct identification of $\mathcal{T}$, as the following correspondence implies:

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

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.[2] 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 $G(\mathcal{S}) \equiv \mathcal{T}$.*

[2] 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.

$$\begin{aligned} \text{Mother} &\equiv \text{Woman} \sqcap \exists \text{hasChild}.\top \\ \text{Father} &\equiv \text{Man} \sqcap \exists \text{hasChild}.\top \\ \text{Father\_of\_son} &\equiv \text{Father} \sqcap \exists \text{hasChild}.\text{Man} \end{aligned}$$

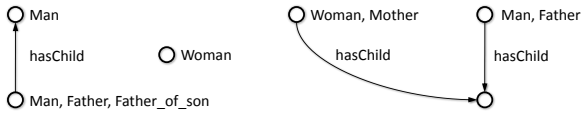

Figure 1: A sample TIP with an $\mathcal{EL}$ TBox (above, where $C \equiv D$ abbreviates $C \sqsubseteq D$ and $D \sqsubseteq C$) and a finite learning set (below).

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. Intuitively, any finite collection of data, structured with respect to some implicitly adopted ontology, can be seen as a potentially instructive learning set, as presented in an example in Figure 1. 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 [Shapiro, 1981], 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 $\mathcal{T}^{\models}$ to be the set of all concept inclusions in $\mathcal{L}$ that are not entailed by $\mathcal{T}$, i.e., $\mathcal{T}^{\models} = \{C \sqsubseteq D \text{ in } \mathcal{L} \mid \mathcal{T} \not\models C \sqsubseteq D\}$. The admissibility condition requires that for every $C \sqsubseteq D \in \mathcal{T}^{\models}$, 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 $\mathcal{T}^{\models}$, 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 signatures, such as $\Sigma_{\mathcal{T}}$. Hence, the set $\mathcal{T}^{\models}$ 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 the online technical report [Klarman and Britz, 2015]. 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 $(\mathcal{J}, d) \mapsto (\mathcal{I}, d)$ for every $(\mathcal{I}, d) \in \mathcal{S}(C)$, and 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 $\tau_n \in \mathcal{T}^{\models}$ 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 exists some $n \in \mathbb{N}$, such that $\tau_m \in \mathcal{T}^{\models}$ 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" $\mathcal{T}^{\models}$ 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.

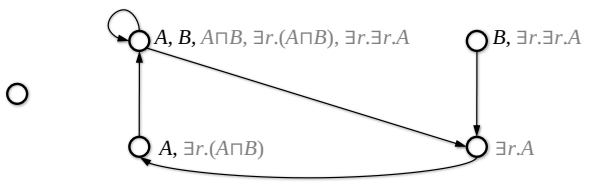

Figure 2: A finite learning set for an $\mathcal{EL}^{\text{rhs}}$ TBox $\{A \sqsubseteq \exists r.(A \sqcap B), B \sqsubseteq \exists r.\exists r.A\}$ (all arrows represent r-relations). The figure includes type contents (in grey), as defined in the proof of Theorem 2.

We adopt the technique of so-called *types*, known from the area of modal logics [Pratt, 1979]. Types are finite abstractions of possible individuals in the interpretation domain, out of which arbitrary models can be constructed. Let $\mathit{con}(\mathcal{T})$ be the set of all concepts (and all their subconcepts) occurring in $\mathcal{T}$. A *type* over $\mathcal{T}$ is a set $t \subseteq \mathit{con}(\mathcal{T})$, such that $C \sqcap D \in t$ *iff* $C \in t$ and $D \in t$, for every $C \sqcap D \in \mathit{con}(\mathcal{T})$. 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 $S \subseteq \mathit{con}(\mathcal{T})$, 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 2 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. ❑

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 3.

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:

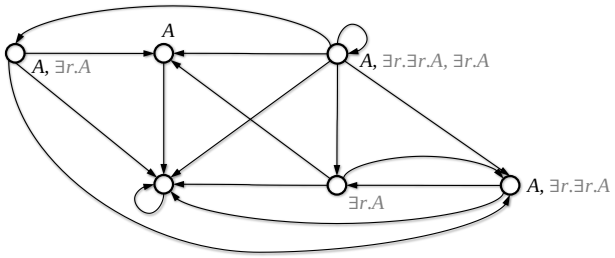

Figure 3: A finite learning set for an $\mathcal{EL}^{\text{lhs}}$ TBox $\{\exists r.\exists r.A \sqsubseteq A\}$ (all arrows represent r-relations). The figure includes type contents (in grey), as defined in the proof of Theorem 3.

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

5 Learning Algorithms

In this section we devise basic learning algorithms that correctly identify $\mathcal{EL}^{\text{lhs}}$ and $\mathcal{EL}^{\text{rhs}}$ TBoxes in admissible TIPs based on finite learning sets. Since $\mathcal{T}^{\not\models}$ can be in general still 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 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

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 $\mathcal{H}$

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

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 combination of individuals is about to be visited for the second time on the same branch it is skipped, as the cycle is evidently detected. 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. Theorem 5 demonstrates the correctness of the whole learning procedure.

Definition 5 ($\mathcal{EL}^{\text{rhs}}$ hypothesis TBox) *Let $(\mathcal{T}, \mathcal{S})$ be an admissible TIP for $\mathcal{T}$ in $\mathcal{EL}^{rhs}$ over the signature $\Sigma_{\mathcal{T}}$. 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 $\mathcal{S}(C) = \emptyset$.*

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 $\mathcal{T}$.*

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: $\quad$ **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: $\quad\quad \Delta^{\mathcal{J}} := \Delta^{\mathcal{J}} \cup \{g(e_0, \ldots, e_n)\}$, for a "fresh" function symbol g
7: $\quad\quad r^{\mathcal{J}} := r^{\mathcal{J}} \cup \{(f(d_0, \ldots, d_n), g(e_0, \ldots, e_n))\}$
8: $\quad$ **end if**
9: **end for**
10: **for** every $f(d_0, \ldots, d_n) \in \Delta^{\mathcal{J}}$, $A \in N_C$ **do**
11: $\quad$ **if** $d_i \in A^{\mathcal{I}_i}$ for every $0 \leq i \leq n$ **then**
12: $\quad\quad A^{\mathcal{J}} := A^{\mathcal{J}} \cup \{f(d_0, \ldots, d_n)\}$
13: $\quad$ **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.

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 related learning strategy should also be applicable in the context of $\mathcal{EL}^{\text{lhs}}$.

6 Related Work

An alternative approach to learning DL TBoxes, based on Angluin's model of learning from entailment [Angluin, 1988], was recently introduced by Konev et al. [Konev *et al.*, 2014]. There, the learner identifies the TBox by posing two types of queries: entailment ("$\mathcal{T} \models C \sqsubseteq D$?") and equivalence ("$\mathcal{H} \equiv \mathcal{T}$? If no, then return a positive or a negative counterexample"). The authors study polynomial learnability and define corresponding algorithms for $\mathcal{EL}^{\text{lhs}}$ and $\mathcal{EL}^{\text{rhs}}$, while for $\mathcal{EL}$ they show that such polynomial algorithm does not exist. Apart from the obvious differences in the motivation underlying both learning models (unsupervised learning from data vs. learning by queries from an expert), there are also some strong formal connections. Essentially, given a finite learning set in an admissible TIP, a learner from interpretations can autonomously answer arbitrary entailment queries, thus effectively simulating the entailment oracle. However, the learner does not have by default access to the equivalence

oracle. Once such oracle is included, as done in our Algorithm 1, the learning power of both learners becomes comparable (note that with some smart heuristic our learner can find a positive or negative counterexample whenever the oracle gives a negative answer). In this sense, our Theorem 4 should be also indirectly derivable from the results by Konev et al. However, our stronger result for $\mathcal{EL}^{\text{rhs}}$ in Theorem 5 demonstrates that, at least in some cases, the learner from interpretations is able to succeed without employing the equivalence oracle, which is essential to the other approach.

Less directly, our work is also related to various contributions on learnability of different types of formal structures from data, e.g.: first-order theories from facts [Shapiro, 1981], finite automata descriptions from observations [Pitt, 1989], logic programs from interpretations [De Raedt and Lavrač, 1993; De Raedt, 1994]. In the area of DLs, a few learning scenarios have been formally addressed, concerned largely with learning concept descriptions via different learning operators [Straccia and Mucci, 2015; Lehmann and Hitzler, 2008; Fanizzi *et al.*, 2008; Cohen and Hirsh, 1994] and applications of formal concept analysis techniques to automated generation of DL axioms from data [Baader *et al.*, 2007; Distel, 2011].

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 a lot of 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 $\mathcal{EL}^{\text{lhs}}$ 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

[Angluin, 1988] Dana Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1988.

[Baader *et al.*, 2003] Franz Baader, Diego Calvanese, Deborah L. Mcguinness, Daniele Nardi, and Peter F. Patel-Schneider. *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, 2003.

[Baader *et al.*, 2005] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of IJCAI-05*, 2005.

[Baader *et al.*, 2007] Franz Baader, Bernhard Ganter, Baris Sertkaya, and Ulrike Sattler. Completing description logic knowledge bases using formal concept analysis. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI-07)*, 2007.

[Cohen and Hirsh, 1994] William W. Cohen and Haym Hirsh. Learning the classic description logic: Theoretical and experimental results. In *Proc. of Principles of Knowledge Representation and Reasoning (KR-94)*, 1994.

[De Raedt and Lavrač, 1993] Luc De Raedt and Nada Lavrač. The many faces of inductive logic programming. In *Methodologies for Intelligent Systems*, pages 435–449. 1993.

[De Raedt, 1994] Luc De Raedt. First order jk-clausal theories are PAC-learnable. *Artificial Intelligence*, 70:375–392, 1994.

[Distel, 2011] Felix Distel. *Learning Description Logic Knowledge Bases from Data using Methods from Formal Concept Analysis*. PhD thesis, TU Dresden, 2011.

[Fanizzi *et al.*, 2008] Nicola Fanizzi, Claudia dAmato, and Floriana Esposito. DL-FOIL concept learning in description logics. *Inductive Logic Programming*, pages 107–121, 2008.

[Hoekstra, 2010] Rinke Hoekstra. The knowledge reengineering bottleneck. *Journal of Semantic Web*, 1(1,2):111–115, 2010.

[Klarman and Britz, 2015] Szymon Klarman and Katarina Britz. Towards unsupervised ontology learning from data. Technical report, CAIR, UKZN/CSIR Meraka. `http://klarman.synthasite.com/resources/KlaBri-DARe15.pdf`, 2015.

[Konev *et al.*, 2014] Boris Konev, Carsten Lutz, Ana Ozaki, and Frank Wolter. Exact learning of lightweight description logic ontologies. In *Proc. of Principles of Knowledge Representation and Reasoning (KR-14)*, 2014.

[Lehmann and Hitzler, 2008] Jens Lehmann and Pascal Hitzler. A refinement operator based learning algorithm for the $\mathcal{ALC}$ description logic. In Hendrik Blockeel, Jan Ramon, Jude Shavlik, and Prasad Tadepalli, editors, *Inductive Logic Programming*, pages 147–160. Springer Berlin Heidelberg, 2008.

[Lutz *et al.*, 2010] Carsten Lutz, Robert Piro, and Frank Wolter. Enriching $\mathcal{EL}$ concepts with greatest fixpoints. In *Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pages 41–46. IOS Press, 2010.

[Maedche and Staab, 2004] Alexander Maedche and Steffen Staab. Ontology learning. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 173–189. Springer, 2004.

[Pitt, 1989] Leonard Pitt. Inductive inference, DFAs, and computational complexity. In Klaus P. Jantke, editor, *Analogical and Inductive Inference*, volume 397 of *Lecture Notes in Computer Science*, pages 18–44. Springer Berlin Heidelberg, 1989.

[Pratt, 1979] V.R. Pratt. Models of program logics. In *Proc. of Foundations of Computer Science (FOCS-79)*, 1979.

[Shapiro, 1981] Ehud Y. Shapiro. Inductive inference of theories from facts. In *Computational Logic: Essays in Honor of Alan Robinson (1991)*. MIT Press, 1981.

[Straccia and Mucci, 2015] Umberto Straccia and Matteo Mucci. pFOIL-DL: Learning (fuzzy) $\mathcal{EL}$ concept descriptions from crisp OWL data using a probabilistic ensemble estimation. In *Proc. of the 30th Annual ACM Symposium on Applied Computing (SAC-15)*. ACM, 2015.