Actively Learning Concepts and Conjunctive Queries under $\mathcal{EL}^r$-Ontologies

Maurice Funk[1] , **Jean Christoph Jung**[2] and **Carsten Lutz**[1]
[1]University of Bremen
[2]University of Hildesheim
mfunk@uni-bremen.de, jungj@uni-hildesheim.de, clu@uni-bremen.de

Abstract

We consider the problem to learn a concept or a query in the presence of an ontology formulated in the description logic $\mathcal{EL}^r$, in Angluin's framework of active learning that allows the learning algorithm to interactively query an oracle (such as a domain expert). We show that the following can be learned in polynomial time: (1) $\mathcal{EL}$-concepts, (2) symmetry-free $\mathcal{ELI}$-concepts, and (3) conjunctive queries (CQs) that are chordal, symmetry-free, and of bounded arity. In all cases, the learner can pose to the oracle membership queries based on ABoxes and equivalence queries that ask whether a given concept/query from the considered class is equivalent to the target. The restriction to bounded arity in (3) can be removed when we admit unrestricted CQs in equivalence queries. We also show that $\mathcal{EL}$-concepts are not polynomial query learnable in the presence of $\mathcal{ELI}$-ontologies.

1 Introduction

In logic based knowledge representation, a significant bottleneck is the construction of logical formulas such as description logic (DL) concepts, queries, and ontologies, as it is laborious and expensive. This is particularly true if the construction involves multiple parties because logic expertise and domain knowledge are not in the same hands. Angluin's model of exact learning, a form of active learning, is able to support the construction of logical formulas in terms of a game-like collaboration between a learner and an oracle [Angluin, 1987b; Angluin, 1987a]. Applied in knowledge representation, the learner can be a logic expert and the oracle a domain expert that is interactively queried by the learner. Alternatively, the oracle can take other forms such as a set of labeled data examples that in some way represents the formula to be learned. The aim is to find an algorithm that, when executed by the learner, constructs the desired formula in polynomial time even when the oracle is not able to provide most informative answers. Landmark results from active learning state that such algorithms exist for learning propositional Horn formulas and finite automata [Angluin *et al.*, 1992; Angluin, 1987a].

The aim of this paper is to study active learning of *DL concepts* and of *conjunctive queries (CQs)* in the presence of an ontology. Concepts are the main building block of ontologies [Baader *et al.*, 2017] and learning them is important for ontology engineering. CQs are very prominent in ontology-mediated querying where data stored in an ABox is enriched with an ontology [Bienvenu *et al.*, 2014]. We concentrate on the $\mathcal{EL}$ family of DLs which underlies the OWL EL profile of the OWL 2 ontology language [Krötzsch, 2012] and is frequently used in biomedical ontologies such as SNOMED CT. We consider ontologies formulated in the DLs $\mathcal{EL}^r$ and $\mathcal{ELI}$ where $\mathcal{EL}^r$ extends $\mathcal{EL}$ with range restrictions and $\mathcal{ELI}$ extends $\mathcal{EL}^r$ with inverse roles. In both DLs, concepts can be viewed as a tree-shaped conjunctive query, and from now on we shall treat them as such. In fact, it is not uncommon to use concepts as queries in ontology-mediated querying, which provides an additional motivation for learning them.

We now describe the learning protocol in detail. It is an instance of Angluin's model, which we do not repeat here in full generality. The aim is to learn a target CQ $q_T(\bar{x})$ in the presence of an ontology $\mathcal{O}$. The learner and the oracle both know and agree on the ontology $\mathcal{O}$, the arity of q_T, and the concept and role names that are available for constructing q_T; we assume that all concept and role names in $\mathcal{O}$ can be used also in q_T. The learner can ask two types of queries to the oracle. In a *membership query*, the learner provides an ABox $\mathcal{A}$ and a candidate answer $\bar{a}$ and asks whether $\mathcal{A}, \mathcal{O} \models q_T(\bar{a})$; the oracle faithfully answers "yes" or "no". In an *equivalence query*, the learner provides a hypothesis CQ q_H and asks whether q_H is equivalent to q_T under $\mathcal{O}$; the oracle answers "yes" or provides a counterexample, that is, an ABox $\mathcal{A}$ and tuple $\bar{a}$ such that $\mathcal{A}, \mathcal{O} \models q_T(\bar{a})$ and $\mathcal{A}, \mathcal{O} \not\models q_H(\bar{a})$ (*positive counterexample*) or vice versa (*negative counterexample*). When we learn a restricted class of CQs such as $\mathcal{EL}$-concepts, we assume that only CQs from that class are admitted in equivalence queries. We are then interested in whether there is a learning algorithm that constructs $q_T(\bar{x})$, up to equivalence under $\mathcal{O}$, such that at any given time, the running time of the algorithm is bounded by a polynomial in the sizes of q_T, of $\mathcal{O}$, and of the largest counterexample given by the oracle so far. This is called *polynomial time learnability*. A weaker requirement is *polynomial query learnability* where only the sum of the sizes of the queries posed to the oracle up to the current time point has to be bounded by such a polynomial.

Our main results are that the following can be learned in polynomial time under $\mathcal{EL}^r$-ontologies: (1) $\mathcal{EL}$-concepts, (2) $\mathcal{ELI}$-concepts that are symmetry-free, and (3) CQs that are chordal, symmetry-free, and of bounded arity. In Point (2), symmetry-freeness means that there is no subconcept of the form $\exists r.(C \sqcap \exists r^-.D)$ with r a role name, a condition that has recently been introduced in [Jung *et al.*, 2020], in a slightly less general form where r can also be an inverse role. In Point (3), chordal means that every cycle of length at least four that contains at least one quantified variable has a chord and symmetry-free means that the CQ contains no atoms $r(x_1, y), r(x_2, y)$ such that $x_1 \neq x_2$, y is a quantified variable, neither $r(x_1, y)$ nor $r(x_2, y)$ occur on a cycle, and there is no atom $s(z, z)$ for any $z \in \{x_1, x_2, y\}$. An analysis of well-known benchmarks for ontology-mediated querying suggests that the resulting class CQ^{csf} of CQs is sufficiently general to include many relevant CQs that occur in practical applications. Our proofs crucially rely on the use of a finite version of the universal model that is specifically tailored to the class CQ^{csf}. We also show that the restriction to bounded arity can be removed from Point (3) when we admit unrestricted CQs as the argument to equivalence queries. Proving this requires very substantial changes to the learning algorithm.

In addition, we prove several negative results. First, we show that none of the classes of CQs in Points (1) to (3) can be learned under $\mathcal{EL}$-ontologies using only membership queries or only equivalence queries (unless P = NP in the latter case). Note that polynomial time learning with only membership queries is important because it is related to whether CQs can be characterized up to equivalence using only polynomially many data examples [ten Cate and Dalmau, 2020]. We also show the much more involved result that none of the classes of CQs in Points (1) to (3) is polynomial query learnable under $\mathcal{ELI}$-ontologies. Note that while polynomial time learnability cannot be expected because subsumption in $\mathcal{ELI}$ is ExpTime-complete, there could well have been a polynomial time learning algorithm with access to an oracle (in the classical sense) for subsumption/query containment under $\mathcal{ELI}$-ontologies that attains polynomial query learnability. Our result rules out this possibility. The appendix with proof details is available in [Funk *et al.*, 2021].

Related work. Learning $\mathcal{EL}$-ontologies, rather than concepts or queries, was studied in [Konev *et al.*, 2018; Konev *et al.*, 2016]. It turns out that $\mathcal{EL}$-ontologies are not polynomial time learnable while certain fragments thereof are. In contrast, we attain polynomial time learnability also under unrestricted $\mathcal{EL}$-ontologies. See also the surveys [Lehmann and Völker, 2014; Ozaki, 2020] and [Ozaki *et al.*, 2020] for a variation less related to the current work. It has been shown in [ten Cate *et al.*, 2013; ten Cate *et al.*, 2018] that unions of CQs (UCQs) are polynomial time learnable, and the presented algorithm can be adapted to CQs. Active learning of CQs with only membership queries is considered in [ten Cate and Dalmau, 2020] where among other results it is shown that $\mathcal{ELI}$-concepts can be learned in polynomial time with only membership queries when the ontology is empty. PAC learnability of concepts formulated in the DL CLASSIC, without ontologies, was studied in [Cohen and Hirsh, 1994b; Cohen and Hirsh, 1994a; Frazier and Pitt, 1996].

2 Preliminaries

Concepts and Ontologies. Let $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$ be countably infinite sets of *concept names*, *role names*, and *individual names*, respectively. A *role* R takes the form r or r^- where r is a role name and r^- is called an *inverse role*. If $R = s^-$ is an inverse role, then R^- denotes the role name s. An $\mathcal{ELI}$*-concept* is formed according to the syntax rule

$$C, D ::= \top \mid A \mid C \sqcap D \mid \exists R.C$$

where A ranges over $\mathsf{N_C}$ and R over roles. An $\mathcal{EL}$*-concept* is an $\mathcal{ELI}$-concept that does not use inverse roles.

An $\mathcal{ELI}$*-ontology* $\mathcal{O}$ is a finite set of *concept inclusions (CIs)* $C \sqsubseteq D$ where C and D range over $\mathcal{ELI}$-concepts. An $\mathcal{EL}^r$*-ontology* is an $\mathcal{ELI}$-ontology where inverse roles occur only in the form of *range restrictions* $\exists r^-.\top \sqsubseteq C$ with C an $\mathcal{EL}$-concept. Note that *domain restrictions* $\exists r.\top \sqsubseteq C$ can be expressed already in $\mathcal{EL}$. An $\mathcal{EL}$-ontology is an $\mathcal{ELI}$-ontology that does not use inverse roles. An $\mathcal{EL}^r$-ontology is in *normal form* if all CIs in it are of one of the forms

$$A_1 \sqcap A_2 \sqsubseteq A,\ A_1 \sqsubseteq \exists r.A_2,\ \exists r.A_1 \sqsubseteq A_2,\ \exists r^-.\top \sqsubseteq A$$

where A, A_1, A_2 are concept names or $\top$. An *ABox* $\mathcal{A}$ is a finite set of *concept assertions* $A(a)$ and *role assertions* $r(a, b)$ where $A \in \mathsf{N_C} \cup \{\top\}$, $r \in \mathsf{N_R}$, and $a, b \in \mathsf{N_I}$. We use $\mathsf{ind}(\mathcal{A})$ to denote the set of individual names that are used in $\mathcal{A}$ and may write $r^-(a, b)$ in place of $r(b, a)$. An ABox is a *ditree* if the directed graph $(\mathsf{ind}(\mathcal{A}), \{(a, b) \mid r(a, b) \in \mathcal{A}\})$ is a tree and there are no multi-edges, that is, $r(a, b), s(a, b) \in \mathcal{A}$ implies $r = s$.

The semantics is defined as usual in terms of *interpretations* $\mathcal{I}$, which we define to be a (possibly infinite and) non-empty set of concept and role assertions. We use $\Delta^{\mathcal{I}}$ to denote the set of individual names in $\mathcal{I}$, define $A^{\mathcal{I}} = \{a \mid A(a) \in \mathcal{I}\}$ for all $A \in \mathsf{N_C}$, and $r^{\mathcal{I}} = \{(a, b) \mid r(a, b) \in \mathcal{I}\}$ for all $r \in \mathsf{N_R}$. The extension $C^{\mathcal{I}}$ of $\mathcal{ELI}$-concepts C is then defined as usual [Baader *et al.*, 2017]. This definition of interpretation is slightly different from the usual one, but equivalent; its virtue is uniformity as every ABox is a (finite) interpretation. An interpretation $\mathcal{I}$ *satisfies* a CI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, and a (concept or role) assertion α if $\alpha \in \mathcal{I}$ or α has the form $\top(a)$. We say that $\mathcal{I}$ is a *model* of an ontology/ABox if it satisfies all concept inclusions/assertions in it and write $\mathcal{O} \models C \sqsubseteq D$ if every model of the ontology $\mathcal{O}$ satisfies the CI $C \sqsubseteq D$.

A *signature* is a set of concept and role names, uniformly referred to as *symbols*. For any syntactic object O such as an ontology or an ABox, we use $\mathsf{sig}(O)$ to denote the symbols used in O and $||O||$ to denote the *size* of O, that is, the length of a word representation of O in a suitable alphabet.

CQs and Homomorphisms. A *conjunctive query (CQ)* takes the form $q(\bar{x}) \leftarrow \varphi(\bar{x}, \bar{y})$ where φ is a conjunction of *concept atoms* $A(x)$ and *role atoms* $r(x, y)$ with $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. We may write $r^-(x, y)$ in place of $r(y, x)$. Note that the tuple $\bar{x}$ used in the *head* $q(\bar{x})$ of the CQ may contain repeated occurrences of variables. When we do not want to make the *body* $\varphi(\bar{x}, \bar{y})$ explicit, we may denote $q(\bar{x}) \leftarrow \varphi(\bar{x}, \bar{y})$ simply with $q(\bar{x})$. We refer to the variables in $\bar{x}$ as the *answer variables* of q. and to the variables in $\bar{y}$ as the *quantified variables*. When we are not interested in order and multiplicity, we treat $\bar{x}$ and $\bar{y}$ as sets of variables. We use $\mathsf{var}(q)$ to denote the set of all variables in $\bar{x}$ and $\bar{y}$. The *arity* of q is the length of tuple $\bar{x}$ and q is *Boolean* if it has arity zero. Every CQ $q(\bar{x}) \leftarrow \varphi(\bar{x}, \bar{y})$ gives rise to an ABox (and thus interpretation) $\mathcal{A}_q$ obtained from $\varphi(\bar{x}, \bar{y})$ by viewing variables as individual names and atoms as assertions. A CQ is a *ditree* if $\mathcal{A}_q$ is.

A *homomorphism* h from interpretation $\mathcal{I}_1$ to interpretation $\mathcal{I}_2$ is a mapping from $\Delta^{\mathcal{I}_1}$ to $\Delta^{\mathcal{I}_2}$ such that $d \in A^{\mathcal{I}_1}$ implies $h(d) \in A^{\mathcal{I}_2}$ and $(d, e) \in r^{\mathcal{I}_1}$ implies $(h(d), h(e)) \in r^{\mathcal{I}_2}$. For $\bar{d}_i$ a tuple over $\Delta^{\mathcal{I}_i}$, $i \in \{1, 2\}$, we write $\mathcal{I}_1, \bar{d}_1 \to \mathcal{I}_2, \bar{d}_2$ if there is a homomorphism h from $\mathcal{I}_1$ to $\mathcal{I}_2$ with $h(\bar{d}_1) = \bar{d}_2$. With a homomorphism from a CQ q to an interpretation $\mathcal{I}$, we mean a homomorphism from $\mathcal{A}_q$ to $\mathcal{I}$.

Let $q(\bar{x}) \leftarrow \varphi(\bar{x}, \bar{y})$ be a CQ and $\mathcal{I}$ an interpretation. A tuple $\bar{d} \in (\Delta^{\mathcal{I}})^{|\bar{x}|}$ is an *answer to* q *on* $\mathcal{I}$, written $\mathcal{I} \models q(\bar{d})$, if there is a homomorphism h from q to $\mathcal{I}$ with $h(\bar{x}) = \bar{d}$. Now let $\mathcal{O}$ be an $\mathcal{ELI}$-ontology and $\mathcal{A}$ an ABox. A tuple $\bar{a} \in \mathsf{ind}(\mathcal{A})^{|\bar{x}|}$ is an *answer to* q *on* $\mathcal{A}$ *under* $\mathcal{O}$, written $\mathcal{A}, \mathcal{O} \models q(\bar{a})$ if $\bar{a}$ is an answer to q on every model of $\mathcal{O}$ and $\mathcal{A}$.

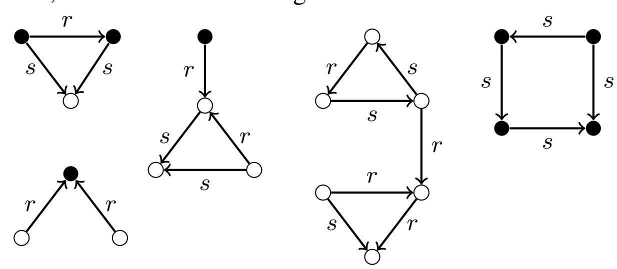
For q_1 and q_2 CQs of the same arity n and $\mathcal{O}$ an $\mathcal{ELI}$-ontology, we say that q_1 is *contained* in q_2 under $\mathcal{O}$, written $q_1 \subseteq_{\mathcal{O}} q_2$, if for all ABoxes $\mathcal{A}$ and $\bar{a} \in \mathsf{ind}(\mathcal{A})^n$, $\mathcal{A}, \mathcal{O} \models q_1(\bar{a})$ implies $\mathcal{A}, \mathcal{O} \models q_2(\bar{a})$. We call q_1 and q_2 *equivalent* under $\mathcal{O}$, written $q_1 \equiv_{\mathcal{O}} q_2$, if $q_1 \subseteq_{\mathcal{O}} q_2$ and $q_2 \subseteq_{\mathcal{O}} q_1$.

Every $\mathcal{ELI}$-concept can be viewed as a unary tree-shaped CQ in an obvious way. For example, the $\mathcal{EL}$-concept $A \sqcap \exists s.\top \sqcap \exists r.B$ yields the CQ $q(x) \leftarrow A(x) \wedge s(x, y) \wedge r(x, z) \wedge B(z)$. We use ELQ to denote the class of all $\mathcal{EL}$-concepts viewed as a CQ, and likewise for ELIQ and $\mathcal{ELI}$-concepts.

Important Classes of CQs. We next define a class of CQs that we show later to admit polynomial time learnability under $\mathcal{EL}^r$-ontologies, one of the main results of this paper. Let $\mathcal{A}$ be an ABox. A *path* in $\mathcal{A}$ from a to b is a sequence $p = R_0(a_0, a_1), \ldots, R_{n-1}(a_{n-1}, a_n) \in \mathcal{A}$, $n \geq 0$, such that $a_0 = a$ and $a_n = b$. We say that p is a *cycle of length* n if $a_0 = a_n$, all assertions in p are distinct, and all of $a_0, \ldots, a_{n-1}$ are distinct. A *chord* of cycle p is an assertion $R(a_i, a_j)$ with $0 \leq i, j < n-1$ and $i \notin \{j, j-1 \mod n, j+1 \mod n\}$. A cycle in a CQ q is a cycle in $\mathcal{A}_q$. With CQ^{csf}, we denote the class of CQs $q(\bar{x}) \leftarrow \varphi(\bar{x}, \bar{y})$ that are

1. *chordal*, that is, every cycle $R_0(x_0, x_1), \ldots, R_{n-2}(x_{n-2}, x_{n-1})$ in q of length at least four that contains at least one quantified variable has a chord;
2. *symmetry-free*, that is, if φ contains atoms $r(y_1, x), r(y_2, x)$ with $y_1 \neq y_2$, then x is an answer variable or one of the atoms occurs on a cycle or φ contains an atom $s(z, z)$ for some $z \in \{x, y_1, y_2\}$.

In Point 2, r is a role name and thus there are no restrictions on 'inverse symmetries': φ may contain atoms $r(x, y_1), r(x, y_2)$ with x a quantified variable and none of the atoms occurring on a cycle and no reflexive loops present. Note that CQ^{csf} contains all CQs without quantified variables (also called *full* CQs), all ELQs, and all ELIQs obtained from $\mathcal{ELI}$-concepts that are *symmetry-free*, that is, that do not contain a subconcept of the form $\exists r.(C \sqcap \exists r^-.D)$ with r a role name. We denote the latter class with ELIQ^{sf}. CQ^{csf} also includes all CQs obtained from such ELIQs by choosing a set of variables and making them answer variables. Note that CQs from CQ^{csf} need not be connected, in fact CQ^{csf} is closed under disjoint union. Every CQ whose graph is a clique or a k-tree (a maximal graph of treewidth k) with $k > 1$ is in CQ^{csf}. Some concrete examples for CQs in CQ^{csf} are given below, filled circles indicating answer variables:



We believe that CQ^{csf} includes many relevant CQs that occur in practical applications. To substantiate this, we have analyzed the 65 queries that are part of three widely used benchmarks for ontology-mediated querying, namely Fishmark, $\text{LUBM}^{\exists}$, and NPD [Bail *et al.*, 2012; Lutz *et al.*, 2013; Lanti *et al.*, 2015]. We found that more than 85% of the queries fall into CQ^{csf} while less than 5% fall into ELIQ^{sf}.

Universal Models. Let $\mathcal{A}$ be an ABox and $\mathcal{O}$ an $\mathcal{EL}^r$-ontology. The *universal model of* $\mathcal{A}$ *and* $\mathcal{O}$, denoted $\mathcal{U}_{\mathcal{A},\mathcal{O}}$, is the interpretation obtained by starting with $\mathcal{A}$ and then 'chasing' with the CIs in the ontology which adds (potentially infinite) ditrees below every $a \in \mathsf{ind}(\mathcal{A})$. The formal definition is in the appendix. The model is universal in that $\mathcal{U}_{\mathcal{A},\mathcal{O}} \models q(\bar{a})$ iff $\mathcal{A}, \mathcal{O} \models q(\bar{a})$ for all CQs $q(\bar{x})$ and tuples $\bar{a} \in \mathsf{ind}(\mathcal{A})^{|\bar{x}|}$. It can be useful to represent universal models in a finite way, as for example in the combined approach to ontology-mediated querying [Lutz *et al.*, 2009]. Here, we introduce a finite representation that is tailored towards our class CQ^{csf}.

The *3-compact model* $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ of $\mathcal{A}$ and $\mathcal{O}$ is defined as follows. Let $\mathsf{sub}(\mathcal{O})$ be the set of all concepts in $\mathcal{O}$, closed under subconcepts. $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ uses the individual names from $\mathcal{A}$ as well as individual names of the form $c_{a,i,r,C}$ where $a \in \mathsf{ind}(\mathcal{A})$, $0 \leq i \leq 4$, r is a role name from $\mathcal{O}$, and $C \in \mathsf{sub}(\mathcal{O})$. For every role name r, we use C_r to denote the conjunction over all C such that $\exists r^-.\top \sqsubseteq C \in \mathcal{O}$, and $\top$ if the conjunction is empty. Let $i \oplus 1$ be short for $(i \mod 4) + 1$. Define

$$\begin{aligned} \mathcal{C}^3_{\mathcal{A},\mathcal{O}} := {} & \mathcal{A} \cup \{A(a) \mid \mathcal{A}, \mathcal{O} \models A(a)\} \cup \\ & \{A(c_{a,i,r,C}) \mid \mathcal{O} \models C \sqcap C_r \sqsubseteq A\} \cup \\ & \{r(a, c_{a,0,r,C}) \mid \mathcal{A}, \mathcal{O} \models \exists r.C(a)\} \cup \\ & \{r(c_{a,i,s,C}, c_{a,i\oplus 1,r,C'}) \mid \mathcal{O} \models C \sqcap C_s \sqsubseteq \exists r.C'\}. \end{aligned}$$

There is a homomorphism from $\mathcal{U}_{\mathcal{A},\mathcal{O}}$ to $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ that is the identity on $\mathsf{ind}(\mathcal{A})$, but in general not vice versa. Nevertheless, $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ is universal for CQ^{csf}.

Lemma 1. *Let $\mathcal{A}$ be an ABox and $\mathcal{O}$ an $\mathcal{EL}^r$-ontology. Then $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ is a model of $\mathcal{A}$ and $\mathcal{O}$ such that for every CQ $q(\bar{x}) \in$ CQ^{csf} and $\bar{a} \in \mathsf{ind}(\mathcal{A})^{|\bar{x}|}$, $\mathcal{C}^3_{\mathcal{A},\mathcal{O}} \models q(\bar{a})$ iff $\mathcal{A}, \mathcal{O} \models q(\bar{a})$.*

$\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ is defined so as to avoid spurious cycles of length at most 3 while larger spurious cycles are irrelevant for CQs that are chordal. This explains the superscript $\cdot^3$ and enables the lemma below. $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ also avoids spurious predecessors connected via different role names. Spurious predecessors connected via the same role name cannot be avoided, but are irrelevant for CQs that are symmetry-free.

Lemma 2. *Every cycle in $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ of length at most three consists only of individuals from* $\mathsf{ind}(\mathcal{A})$.

We also use the direct product $\mathcal{I}_1 \times \mathcal{I}_2$ of interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$, defined in the standard way (see appendix). For tuples of individuals $\bar{a}_i = (a_{i,1}, \ldots, a_{i,n})$, $i \in \{1, 2\}$, we set $\bar{a}_1 \otimes \bar{a}_2 = ((a_{1,1}, a_{2,1}), \ldots, (a_{1,n}, a_{2,n}))$.

3 Learning under $\mathcal{EL}^r$-Ontologies

We establish polynomial time learnability results under $\mathcal{EL}^r$-ontologies for the query classes CQ$^{\text{csf}}$, ELQ, and ELIQ$^{\text{sf}}$. For CQ$^{\text{csf}}$, we additionally have to assume that the arity of CQs to be learned is bounded by a constant or that unrestricted CQs can be used in equivalence queries. When speaking of equivalence queries, we generally imply that the CQs used in such queries must be from the class of CQs to be learned. If this is not the case and unrestricted CQs are admitted in equivalence queries, then we speak of *CQ-equivalence queries*. When using CQ-equivalence queries, the learned representation of the target query is a CQ, but need not necessarily belong to $\mathcal{C}$ (though it is equivalent to a query from $\mathcal{C}$). For $w \geq 0$, let CQ$^{\text{csf}}_w$ be the restriction of CQ$^{\text{csf}}$ to CQs of arity at most w. The following are the main results obtained in this section.

Theorem 1.

1. *ELQ- and ELIQ^{sf}-queries are polynomial time learnable under $\mathcal{EL}^r$-ontologies using membership and equivalence queries;*
2. *for every $w \geq 0$, CQ$^{csf}_w$-queries are polynomial time learnable under $\mathcal{EL}^r$-ontologies using membership and equivalence queries;*
3. *CQ^{csf}-queries are polynomial time learnable under $\mathcal{EL}^r$-ontologies using membership and CQ-equivalence queries.*

Before providing a proof of Theorem 1, we show that both membership and equivalence queries are needed for polynomial learnability. Let AQ$^\wedge$ denote the class of unary CQs of the form $q(x) \leftarrow A_1(x) \wedge \cdots \wedge A_n(x)$, and let a *conjunctive ontology* be an $\mathcal{EL}$-ontology without role names.

Theorem 2.

1. *AQ$^\wedge$-queries are not polynomial query learnable under conjunctive ontologies using only membership queries;*
2. *ELQ-queries are not polynomial time learnable (without ontologies) using only CQ-equivalence queries unless* P = NP.

Note that Points 1 and 2 of Theorem 2 imply the same statements for all relevant query classes, that is, ELQ, ELIQ$^{\text{sf}}$, CQ$^{\text{csf}}$, CQ$^{\text{csf}}_w$ for all $w \geq 1$, and CQ, in place of the classes mentioned in the theorem. In particular, Point 2 implies that unrestricted CQs are not polynomial time learnable with only equivalence queries in the classical setting (without ontologies) unless P = NP, even when only unary and binary relations are admitted, see [Cohen, 1995; Haussler, 1989; Hirata, 2000] for related results. The proof of Point 1 follows basic lower bound proofs for abstract learning problems [Angluin, 1987b]. Point 2 is proved by exploiting connections between active learning and inseparability questions studied in [Funk *et al.*, 2019; Jung *et al.*, 2020; Funk, 2019].

Algorithm 1 Learning queries q_T from ELQ / ELIQ$^{\text{sf}}$ / CQ$^{\text{csf}}_w$ under an $\mathcal{EL}^r$-ontology $\mathcal{O}$.

procedure LEARNCQ
  $q_H(\bar{x}) := \mathsf{refine}(q^\perp(\bar{x}_0))$
  while $q_H \not\equiv_\mathcal{O} q_T$ (equivalence query) **do**
    Let $\mathcal{A}, \bar{a}$ be the positive counterexample returned and let $q'_H(\bar{x}')$ be $\mathcal{C}^3_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ viewed as a CQ with answer variables $\bar{x}' = \bar{x} \otimes \bar{a}$
    $q_H(\bar{x}) := \mathsf{refine}(q'_H(\bar{x}'))$
  return $q_H(\bar{x})$

3.1 Reduction to Normal Form

We show that the ontology under which we learn can w.l.o.g. be assumed to be in normal form. It is well-known that every $\mathcal{EL}^r$-ontology $\mathcal{O}$ can be converted into normal form by introducing fresh concept names [Baader *et al.*, 2017]. We use such a conversion to show that, for the relevant classes of CQs, a polynomial time learning algorithm under $\mathcal{EL}^r$-ontologies in normal form can be converted into a polynomial time learning algorithm under unrestricted $\mathcal{EL}^r$-ontologies. Care has to be exercised as the fresh concept names can occur in membership and equivalence queries. From now on, we thus assume that ontologies are in normal form.

Proposition 1. *Let $\mathcal{Q} \in \{ELQ, ELIQ^{sf}, CQ^{csf}_w \mid w \geq 0\}$. If queries in $\mathcal{Q}$ are polynomial time learnable under $\mathcal{EL}^r$-ontologies in normal form using membership and equivalence queries, then the same is true for unrestricted $\mathcal{EL}^r$-ontologies.*

3.2 Algorithm Overview

We start with proving Points 1 and 2 of Theorem 1. Thus let $\mathcal{Q} \in \{\text{ELQ}, \text{ELIQ}^{\text{sf}}, \text{CQ}^{\text{csf}}_w \mid w \geq 0\}$. The algorithm that establishes polynomial time learnability of queries from $\mathcal{Q}$ under $\mathcal{EL}^r$-ontologies is displayed as Algorithm 1. We next explain some of its details.

Let $\mathcal{O}$ be an $\mathcal{EL}^r$-ontology, Σ a finite signature that contains all symbols in $\mathcal{O}$, and $\mathsf{ar} \leq w$ an arity for the query to be learned with $\mathsf{ar} = 1$ if $\mathcal{Q} \in \{\text{ELQ}, \text{ELIQ}^{\text{sf}}\}$, all known to the learner and the oracle. Further let $q_T(\bar{y}) \in \mathcal{Q}$ be the target query known to the oracle, formulated in signature Σ. The algorithm maintains and repeatedly updates a hypothesis CQ $q_H(\bar{x})$ of arity ar. It starts with the hypothesis

$$q^\perp(\bar{x}_0) \leftarrow \{A(x_0) \mid A \in \Sigma \cap \mathsf{N_C}\} \cup \{r(x_0, x_0) \mid r \in \Sigma \cap \mathsf{N_R}\}$$

where $\bar{x}_0$ contains only the variable x_0, repeated ar times. By construction, $q^\perp \subseteq_\mathcal{O} q$ for all CQs q of arity ar that use only symbols from Σ. Note that $q^\perp \in \mathrm{CQ}^{\mathrm{csf}}_w$ for all w, but $q^\perp$ is neither in ELQ nor in ELIQ$^{\mathrm{sf}}$.

If $q_1(\bar{x}_1), q_2(\bar{x}_2), \ldots$ are the hypotheses constructed during a run of the algorithm, then for all $i \geq 1$:

1. $q_i \in \mathcal{Q}$ and $q_i \subseteq_\mathcal{O} q_T$;
2. $q_i \subseteq_\mathcal{O} q_{i+1}$ and $q_i \not\equiv_\mathcal{O} q_{i+1}$;
3. $|\mathsf{var}(q_i)| \leq |\mathsf{var}(q_T)|$.

Taken together, Points 1 and 2 mean that the hypotheses approximate the target query from below in an increasingly better way and Point 3 is crucial for proving that we must reach q_T after polynomially many steps. The fact that $\mathcal{O}$ is in normal form is used to attain Point 3.

Point 1 also guarantees that the oracle always returns a *positive* counterexample $\mathcal{A}, \bar{a}$ to the equivalence query used to check whether $q_H \not\equiv_\mathcal{O} q_T$ in the while loop. The algorithm extracts the commonalities of $q_H(\bar{x})$ and $\mathcal{A}, \bar{a}$ by means of a direct product with the aim of obtaining a better approximation of the target. The same is done in the case without ontologies [ten Cate *et al.*, 2013] where $\mathcal{A}_{q_H} \times \mathcal{A}$ (viewed as a CQ) is the new hypothesis, but this is not sufficient here as it misses the impact of the ontology. The product $\mathcal{U}_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{U}_{\mathcal{A},\mathcal{O}}$ would work, but need not be finite. So we resort to $\mathcal{C}^3_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ instead, viewed as a CQ $q'_H(\bar{x}')$. This new hypothesis need not belong to $\mathcal{Q}$, so we call the subroutine refine detailed in the subsequent section to convert it into a new hypothesis $q_H(\bar{x}) \in \mathcal{Q}$ such that $q'_H \subseteq_\mathcal{O} q_H \subseteq_\mathcal{O} q_T$. The initial call to refine serves the same purpose as $q^\perp(\bar{x}_0)$ need not be in $\mathcal{Q}$, depending on the choice of $\mathcal{Q}$.

It is not immediately clear that the described approach achieves the containment in Point 2 since $\mathcal{C}^3_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ is potentially too strong as a replacement of $\mathcal{U}_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{U}_{\mathcal{A},\mathcal{O}}$; in particular, there might be cycles in the former product that do not exist in the latter. What saves us, however, is that the CQ q_H constructed by refine belongs to $\mathcal{Q}$ while the models $\mathcal{C}^3_{\mathcal{A}_{q_H},\mathcal{O}}$ and $\mathcal{C}^3_{\mathcal{A},\mathcal{O}}$ are universal for $\mathcal{Q}$ as per Lemma 1.

3.3 The refine Subroutine

The refine subroutine gets as input a CQ $q'_H(\bar{x}')$ that does not need to be in $\mathcal{Q}$, but that satisfies $q'_H \subseteq_\mathcal{O} q_T$. It produces a query $q_H(\bar{x})$ from $\mathcal{Q}$ such that $q'_H \subseteq_\mathcal{O} q_H \subseteq_\mathcal{O} q_T$ and $|\mathsf{var}(q_H)| \leq |\mathsf{var}(q_T)|$. For notational convenience, we prefer to view $q'_H(\bar{x}')$ as a pair $(\mathcal{A}, \bar{a})$ where $\mathcal{A} = \mathcal{A}_{q'_H}$ and $\bar{a} = \bar{x}'$. Let $n_{\max}$ denote the maximum length of a chordless cycle in any query in $\mathcal{Q}$, that is $n_{\max} = 0$ for $\mathcal{Q} \in \{\mathrm{ELQ}, \mathrm{ELIQ}^{\mathrm{sf}}\}$ and $n_{\max} = 3$ for $\mathcal{Q} = \mathrm{CQ}^{\mathrm{csf}}_w$, $w \geq 0$. We shall use the following.

Minimize. Let $\mathcal{B}$ be an ABox and $\bar{b}$ a tuple such that $\mathcal{B}, \mathcal{O} \models q_T(\bar{b})$. Then $\mathsf{minimize}(\mathcal{B}, \bar{b})$ is the ABox $\mathcal{B}'$ obtained from $\mathcal{B}$ by exhaustively applying the following operations:

(1) choose $c \in \mathsf{ind}(\mathcal{B}) \setminus \bar{b}$ and remove all assertions that involve c. Use a membership query to check whether, for the resulting ABox $\mathcal{B}^-$, $\mathcal{B}^-, \mathcal{O} \models q_T(\bar{b})$. If so, proceed with $\mathcal{B}^-$ in place of $\mathcal{B}$.

(2) choose $r(a, b) \in \mathcal{B}$ and use a membership query to check whether $\mathcal{B} \setminus \{r(a, b)\}, \mathcal{O} \models q_T(\bar{b})$. If so, proceed with $\mathcal{B} \setminus \{r(a, b)\}$ in place of $\mathcal{B}$.

The refine subroutine builds a sequence $(\mathcal{B}_1, \bar{b}_1), (\mathcal{B}_2, \bar{b}_2), \ldots$ starting with $(\mathcal{B}_1, \bar{b}_1) = (\mathsf{minimize}(\mathcal{A}, \bar{a}), \bar{a})$ and exhaustively applying the following step:

Expand. Choose a chordless cycle $R_0(a_0, a_1), \ldots, R_{n-1}(a_{n-1}, a_n)$ in $\mathcal{B}_i$ with $n > n_{\max}$ and, in case that $\mathcal{Q} = \mathrm{CQ}^{\mathrm{csf}}_w$, $\{a_0, \ldots, a_{n-1}\} \not\subseteq \bar{b}_i$.[1] Let $\mathcal{B}'_i$ be the ABox obtained by doubling the length of the cycle: start with $\mathcal{B}_i$, introduce copies $a'_0, \ldots, a'_{n-1}$ of $a_0, \ldots, a_{n-1}$, and then

- remove all assertions $R(a_{n-1}, a_0)$;
- add $B(a'_i)$ if $B(a_i) \in \mathcal{B}_i$;
- add $R(a'_i, c)$ if $R(a_i, c) \in \mathcal{B}_i$ with $0 \leq i < n$ and $c \in \mathsf{ind}(\mathcal{B}_i) \setminus \{a_0, \ldots, a_{n-1}\}$;
- add $R(a'_i, a'_j)$ if $R(a_i, a_j) \in \mathcal{B}_i$ with $0 \leq i, j < n$ and $\{i, j\} \neq \{0, n-1\}$;
- add $R(a_{n-1}, a'_0)$ and $R(a'_{n-1}, a_0)$ if $R(a_{n-1}, a_0) \in \mathcal{B}_i$.

A similar construction is used in [Konev *et al.*, 2016]. Let τ_i be the set of tuples $\bar{b}$ obtained from $\bar{b}_i = (b_1, \ldots, b_k)$ by replacing any number of components b_j by b'_j. Use membership queries to identify $\bar{b}_{i+1} \in \tau_i$ with $\mathcal{B}'_i, \mathcal{O} \models q_T(\bar{b}_{i+1})$ and set $\mathcal{B}_{i+1} = \mathsf{minimize}(\mathcal{B}'_i, \bar{b}_{i+1})$. We prove in the appendix that such a $\bar{b}_{i+1}$ always exists and that the Expand step can only be applied polynomially many times. The resulting $(\mathcal{B}_n, \bar{b}_n)$ viewed as a CQ with answer variables $\bar{b}_n$ is chordal, but not necessarily symmetry-free. To establish also the latter, we compute a sequence of ABoxes $\mathcal{B}_n, \mathcal{B}_{n+1}, \ldots$ by exhaustively applying the following step:

Split. Choose $r(a, b), r(c, b) \in \mathcal{B}_i$ such that $b \notin \bar{b}_n$ and neither $r(a, b)$ nor $r(c, b)$ occurs on a cycle. Construct $\mathcal{B}'_i$ by removing $r(a, b)$ from $\mathcal{B}_i$, taking a fresh individual b', and adding $B(b')$ for all $B(b) \in \mathcal{B}_i$ and $S(d, b')$ for all $S(d, b) \in \mathcal{B}_i$ with $S(d, b) \neq r(c, b)$. If $\mathcal{B}'_i, \mathcal{O} \models q_T(\bar{b}_n)$, then $\mathcal{B}_{i+1} = \mathsf{minimize}(\mathcal{B}'_i, \bar{b}_n)$.

We prove in the appendix that only polynomially many applications are possible and that, for $\mathcal{B}_m$ the resulting ABox, $(\mathcal{B}'_m, \bar{b}_n)$ viewed as a CQ is chordal and symmetry-free. Moreover, it is in ELQ if q_T is, and likewise for ELIQ$^{\mathrm{sf}}$. Refine returns this CQ as its result. Note that the running time of refine depends exponentially on ar due to the brute force search for a tuple $\bar{b}_{i+1} \in \tau_i$ in the Expand step.

3.4 Unbounded Arity

To prove the remaining Point 3 of Theorem 1, we have to deal with CQs of unbounded arity and cannot use the refine subroutine presented in Section 3.3. We thus introduce a second version of refine that works rather differently from the previous one. We give an informal description, full details are in the appendix.

Recall that refinement starts with the product $P = \mathcal{C}^3_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{C}^3_{\mathcal{A},\mathcal{O}}$. In Section 3.3, we blow up cycles in P, not distinguishing the ABox part and the existentially generated part of the 3-compact models involved. The second version of refine instead unravels the existentially generated

[1] This is because CQ$^{\mathrm{csf}}$ admits cycles that consist only of answer variables while ELQ and ELIQ$^{\mathrm{sf}}$ do not.

part of the two 3-compact models inside the product P. A full such unraveling would eventually result in $\mathcal{U}_{\mathcal{A}_{q_H},\mathcal{O}} \times \mathcal{U}_{\mathcal{A},\mathcal{O}}$, but we interleave with a Minimize step as in Section 3.3 and thus obtain a finite initial piece thereof. Unlike in the previous version of refine, we do not have to redefine the answer variables at all (but note that they may still change outside of refine when we take the product).

The above suffices for target CQs from $\mathrm{CQ}^{\mathrm{csf}}$ in which every variable is reachable from an answer variable. In the general case, disconnected Boolean components might be present (or emerge during unraveling and minimization) that are never unraveled. To address this, we subsequently apply the original version of refine to such components, avoiding the Splitting step and leaving the already unraveled parts untouched. Note that the exponential blowup in the arity is avoided because the original refine is only applied to Boolean subqueries. However, the resulting queries are not guaranteed to be in $\mathrm{CQ}^{\mathrm{csf}}$. We can thus not rely on Lemma 1 as before which is why we need CQ-equivalence queries.

4 Learning under $\mathcal{ELI}$-Ontologies

When we replace $\mathcal{EL}^r$-ontologies with $\mathcal{ELI}$-ontologies, polynomial time learnability can no longer be expected since containment between ELQs under $\mathcal{ELI}$-ontologies is ExpTime-complete [Baader *et al.*, 2008]. In contrast, polynomial query learnability is not ruled out and in fact it is natural to ask whether there is a polynomial time learning algorithm with access to an oracle (in the classical sense) for query containment under $\mathcal{ELI}$-ontologies. Note that such an algorithm would show polynomial query learnability. We answer this question to the negative and show that polynomial query learnability cannot be attained under $\mathcal{ELI}$-ontologies for any of the query classes considered in this paper. This is a consequence of the following result, which also captures learning of unrestricted CQs.

Theorem 3. *$\mathcal{EL}$-concepts are not polynomial query learnable under $\mathcal{ELI}$-ontologies with membership queries and CQ-equivalence queries.*

For the proof, we use the $\mathcal{ELI}$-ontologies $\mathcal{O}_n$, $n \geq 1$, given in Figure 1. There, $\overline{r} = s$ and $\overline{s} = r$. Every $\mathcal{O}_n$ is associated with a set $\mathcal{H}_n$ of 2^n potential target concepts of the form

$$\exists\sigma_1 \cdots \exists\sigma_n.\exists r^n.A \text{ with } \sigma_1, \ldots, \sigma_n \in \{r, s\}$$

where $\exists r^n$ denotes the n-fold nesting of $\exists r$. The idea of the proof is to show that if there was an algorithm for learning $\mathcal{EL}$-concepts under $\mathcal{ELI}$-ontologies such that, at any given time, the sum of the sizes of all (membership and CQ-equivalence) queries asked to the oracle is bounded by a polynomial $p(n_1, n_2, n_3)$ with n_1 is the size of the target query, n_2 is the size of the ontology, and n_3 is the size of the largest counterexample seen so far, then we can choose n large enough so that the learner needs more than $p(n_1, n_2, n_3)$ queries to distinguish the targets in $\mathcal{H}_n$ under $\mathcal{O}_n$ if the oracle uses a 'sufficiently destructive' strategy to answer the queries. Such a strategy is presented in the appendix, we only give one example that highlights a crucial aspect.

Assume that the learner poses as an equivalence query the $\mathcal{EL}$-concept $C_H = \exists\sigma_1 \cdots \exists\sigma_n.\exists r^n.A$. Then the oracle returns "no" and positive counterexample $\mathcal{A} = \{K_0(a_0), W_1^{\sigma_1}(a_0), \ldots, W_n^{\sigma_n}(a_0)\}$. It is instructive to verify that $\mathcal{A}, \mathcal{O} \models C'_H(a_0)$ for all $C'_H \in \mathcal{H}_n \setminus \{C_H\}$ while $\mathcal{A}, \mathcal{O} \not\models C_H(a_0)$ as this illustrates the use of inverse roles in $\mathcal{O}_n$.

$\top \sqsubseteq \exists r.\top \sqcap \exists s.\top$	
$L_i \sqsubseteq \exists r.L_{i+1} \sqcap \exists s.L_{i+1}$	for $0 \leq i \leq n$
$L_i \sqsubseteq \exists r.L_{i+1}$	for $n \leq i < 2n$
$L_{2n} \sqsubseteq A$	
$\exists\sigma.L_{i+1} \sqsubseteq L_i$	for $\sigma \in \{r, s\}$ and $0 \leq i \leq 2n$
$K_i \sqsubseteq \exists r.(K_{i+1} \sqcap V_{i+1}^r) \sqcap \exists s.(K_{i+1} \sqcap V_{i+1}^s)$	for $\sigma \in \{r, s\}$ and $0 \leq i \leq n$
$K_i \sqcap W_{i+1}^\sigma \sqsubseteq \exists r.K_{i+1}$	for $\sigma \in \{r, s\}$ and $n \leq i < 2n$
$\exists\sigma^-.(K_j \sqcap V_i^{\sigma'}) \sqsubseteq V_i^{\sigma'}$	for $\sigma, \sigma' \in \{r, s\}, 1 \leq i \leq n$, and $i \leq j \leq 2n$
$K_{2n} \sqcap V_i^\sigma \sqcap W_i^{\overline{\sigma}} \sqsubseteq A$	for $\sigma \in \{r, s\}$ and $1 \leq i \leq n$
$\exists\sigma.W_i^{\sigma'} \sqsubseteq W_i^{\sigma'}$	for $\sigma \in \{r, s, r^-, s^-\}$, $\sigma' \in \{r, s\}$, and $1 \leq i \leq n$
$W_i^r \sqcap W_i^s \sqsubseteq L_0$	for $0 \leq i \leq n$
$\exists\sigma.K_{i+1} \sqsubseteq K_i$	for $\sigma \in \{r, s\}$ and $0 \leq i \leq 2n$
$\exists\sigma^-.\top \sqsubseteq U_1^\sigma$	for $\sigma \in \{r, s\}$
$\exists\sigma^-.U_i^{\sigma'} \sqsubseteq U_{i+1}^{\sigma'}$	for $\sigma, \sigma' \in \{r, s\}$ and $1 \leq i < 2n$
$U_i^r \sqcap U_i^s \sqsubseteq D$	for $1 \leq i \leq 2n$
$K_i \sqcap A \sqsubseteq D$	for $0 \leq i < 2n$
$L_i \sqcap A \sqsubseteq D$	for $0 \leq i < 2n$
$L_i \sqcap L_j \sqsubseteq D$	for $n \leq i < j \leq 2n$
$K_i \sqcap K_j \sqsubseteq D$	for $n \leq i < j \leq 2n$
$L_i \sqcap K_j \sqsubseteq D$	for $n \leq i, j \leq 2n$
$\exists\sigma.D \sqsubseteq D$	for $\sigma \in \{r, s, r^-, s^-\}$
$D \sqsubseteq L_0$	

Figure 1: $\mathcal{ELI}$-ontology $\mathcal{O}_n$

5 Conclusion

We conjecture that our results can be extended from $\mathcal{EL}^r$-ontologies to $\mathcal{ELH}^r$-ontologies, thus adding role inclusions. In contrast, we do not know how to learn in polynomial time unrestricted $\mathcal{ELI}$-concepts under $\mathcal{EL}$-ontologies, or symmetry-free CQs under $\mathcal{EL}$-ontologies. We would not be surprised if these indeed turn out not to be learnable in polynomial time. It is an interesting question whether our results can be generalized to symmetry-free CQs that admit chordless cycles of length bounded by a constant larger than three. This would require the use of a different kind of compact universal model.

Acknowledgements

Supported by the DFG Collaborative Research Center 1320 EASE - Everyday Activity Science and Engineering.

References

[Angluin *et al.*, 1992] Dana Angluin, Michael Frazier, and Leonard Pitt. Learning conjunctions of Horn clauses. *Mach. Learn.*, 9:147–164, 1992.

[Angluin, 1987a] Dana Angluin. Learning regular sets from queries and counterexamples. *Inf. Comput.*, 75(2):87–106, 1987.

[Angluin, 1987b] Dana Angluin. Queries and concept learning. *Mach. Learn.*, 2(4):319–342, 1987.

[Baader *et al.*, 2008] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope further. In *Proc. of OWLED*, 2008.

[Baader *et al.*, 2017] Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. *An Introduction to Description Logics*. Cambride University Press, 2017.

[Bail *et al.*, 2012] Samantha Bail, Sandra Alkiviadous, Bijan Parsia, David Workman, Mark van Harmelen, Rafael S. Gonçalves, and Cristina Garilao. Fishmark: A linked data application benchmark. In *Proc. of the Joint Workshop on Scalable and High-Performance Semantic Web Systems*, pages 1–15. CEUR-WS.org, 2012.

[Bienvenu *et al.*, 2014] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive datalog, CSP, and MMSNP. *ACM Trans. Database Syst.*, 39(4):33:1–33:44, 2014.

[Cohen and Hirsh, 1994a] William W. Cohen and Haym Hirsh. The learnability of description logics with equality constraints. *Mach. Learn.*, 17(2-3):169–199, 1994.

[Cohen and Hirsh, 1994b] William W. Cohen and Haym Hirsh. Learning the classic description logic: Theoretical and experimental results. In *Proc. of KR*, pages 121–133. Morgan Kaufmann, 1994.

[Cohen, 1995] William W. Cohen. PAC-learning nonrecursive Prolog clauses. *Artificial Intelligence*, 79(1):1–38, 1995.

[Frazier and Pitt, 1996] Michael Frazier and Leonard Pitt. Classic learning. *Mach. Learn.*, 25(2-3):151–193, 1996.

[Funk *et al.*, 2019] Maurice Funk, Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. Learning description logic concepts: When can positive and negative examples be separated? In *Proc. of IJCAI*, pages 1682–1688, 2019.

[Funk *et al.*, 2021] Maurice Funk, Jean Christoph Jung, and Carsten Lutz. Actively learning concepts and conjunctive queries under $\mathcal{EL}^r$-ontologies. *CoRR*, abs/2105.08326, 2021.

[Funk, 2019] Maurice Funk. Concept-by-Example in $\mathcal{EL}$ Knowledge Bases. Master's thesis, University of Bremen, 2019.

[Haussler, 1989] David Haussler. Learning conjunctive concepts in structural domains. *Mach. Learn.*, 4:7–40, 1989.

[Hirata, 2000] Kouichi Hirata. On the hardness of learning acyclic conjunctive queries. In *Proc. of ALT*, pages 238–251, 2000.

[Jung *et al.*, 2020] Jean Christoph Jung, Carsten Lutz, and Frank Wolter. Least general generalizations in description logic: Verification and existence. In *Proc. of AAAI*, 2020.

[Konev *et al.*, 2016] Boris Konev, Ana Ozaki, and Frank Wolter. A model for learning description logic ontologies based on exact learning. In *Proc. of AAAI*, pages 1008–1015. AAAI Press, 2016.

[Konev *et al.*, 2018] Boris Konev, Carsten Lutz, Ana Ozaki, and Frank Wolter. Exact learning of lightweight description logic ontologies. *J. Mach. Learn. Res.*, 18(201):1–63, 2018.

[Krötzsch, 2012] Markus Krötzsch. OWL 2 profiles: An introduction to lightweight ontology languages. In *Proc. of Reasoning Web*, pages 112–183, 2012.

[Lanti *et al.*, 2015] Davide Lanti, Martín Rezk, Guohui Xiao, and Diego Calvanese. The NPD benchmark: Reality check for OBDA systems. In *Proc. of EDBT*, pages 617–628. OpenProceedings.org, 2015.

[Lehmann and Völker, 2014] Jens Lehmann and Johanna Völker. *Perspectives on Ontology Learning*, volume 18 of *Studies on the Semantic Web*. IOS Press, 2014.

[Lutz *et al.*, 2009] Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic $\mathcal{EL}$ using a relational database system. In *Proc. of IJCAI*, pages 2070–2075, 2009.

[Lutz *et al.*, 2013] Carsten Lutz, Inanç Seylan, David Toman, and Frank Wolter. The combined approach to OBDA: taming role hierarchies using filters. In *Proc. of ISWC*, volume 8218 of *LNCS*, pages 314–330. Springer, 2013.

[Ozaki *et al.*, 2020] Ana Ozaki, Cosimo Persia, and Andrea Mazzullo. Learning query inseparable $\mathcal{ELH}$ ontologies. In *Proc. of AAAI*, pages 2959–2966, 2020.

[Ozaki, 2020] Ana Ozaki. Learning description logic ontologies: Five approaches. where do they stand? *KI - Künstliche Intelligenz*, 2020.

[ten Cate and Dalmau, 2020] Balder ten Cate and Victor Dalmau. Conjunctive queries: Unique characterizations and exact learnability. *CoRR*, abs/2008.06824, 2020.

[ten Cate *et al.*, 2013] Balder ten Cate, Víctor Dalmau, and Phokion G. Kolaitis. Learning schema mappings. *ACM Trans. Database Syst.*, 38(4):28:1–28:31, 2013.

[ten Cate *et al.*, 2018] Balder ten Cate, Phokion G. Kolaitis, Kun Qian, and Wang-Chiew Tan. Active learning of GAV schema mappings. In *Proc. of PODS*, pages 355–368, 2018.