Exploring Controlled English Ontology-Based Data Access

Camilo Thorne, Diego Calvanese

KRDB Research Centre
Free University of Bozen-Bolzano, Italy
{cthorne,calvanese}@inf.unibz.it

1 Introduction

Controlled languages (CLs) are subsets of natural language with minimal ambiguity (lexical, structural, or semantic) tailored to fullfil data management tasks. Recently [5, 11], they have been proposed as a means of providing natural language interfaces to databases (DBs) and to ontology-based data access systems (OBDASs) centered around the W3C standard ontology language OWL[1] (Web Ontology Language) [7, 9]. CL utterances are so-to-speak "compiled" (compositionally translated by a symbolic translation) into a formal query/ontology language expression. They have given rise to a number of applications and implementations [5, 11], among which ACE-OWL [7, 9], which maps to OWL DL and fragments of it (such as OWL-Lite). The formal underpinning of OWL DL is provided by description logics (DLs) [3, 8], in particular, OWL Lite corresponds to the DL $\mathcal{SHIF}$ (Actually $\mathcal{SHIF}(D)$, but we do not consider datatypes).

Given that OBDASs may contain large amounts of data, we are interested in knowing whether querying such systems through CL interfaces scales up with the size of the data. An ODBAS can be modelled by a DL *knowledge base* (KB), whose information is accessed through queries belonging to some fragment of SQL. A fragment that is considered sufficiently expressive but still computationally manageable (since query answering is decidable in significant cases) is that of conjunctive queries [1]. The basic problem in this setting is *(conjunctive) query answering* (QA) [6], which is a form of logical entailment [3]. To evaluate the efficiency of querying through CL interfaces, we focus on the so-called *data complexity* of QA, i.e., the computational complexity of the problem measured in terms of the size of the data only [16].

The optimum lies in **L** data complexity, which is the complexity of answering SQL queries over plain databases. Data complexity beyond **PTime** (i.e., intractable) indicates low scalability of query answering through CL interfaces. The data complexity of query answering in $\mathcal{SHIF}$, and hence in OWL Lite and ACE-OWL Lite (the fragment of ACE that maps to OWL Lite) is known to be **coNP**-complete [12]. Hence, CLs like ACE-OWL do not scale up with data, although they contain fragments that do. This computational behaviour depends on the language constructs they cover.

In this paper we pinpoint (some of) the English constructs and a fortiori of ACE, that give rise to these computational properties. To this effect, we study a CL that maps into $\mathcal{ALCI}$, a DL simpler than $\mathcal{SHIF}$ but with the same data complexity of query answering, and fragments thereof that are either *(i) minimal* w.r.t. intractability or *(ii) maximal* w.r.t. tractability, mirroring Pratt and Third in [13] (cf. also [14]).

[1] http://www.w3.org/TR/owl-ref/

2 QA over Ontologies

In an OBDAS, an ontology provides a conceptual view on the data stored in a database. Ontologies are formally underpinned by DLs [3], which structure the domain of discourse in terms of concepts (representing classes, i.e., sets of objects) and roles (representing binary relations between objects). We are interested in DLs of different expressiveness, ranging from *DL-Lite*$_\sqcap$ [6] to $\mathcal{ALCI}$. In $\mathcal{ALCI}$, *concepts* C and *roles* R are formed according to the following syntax:

$$R \rightarrow P \mid P^- \qquad\qquad C \rightarrow \top \mid A \mid \exists R{:}C \mid \neg C \mid C \sqcap C$$

where A stands for a concept name (a unary predicate), P for a role name (a binary predicate) and P^- for its inverse. The semantics of the concept and role constructors is the standard one for DLs, see [3]. We can enrich the set of $\mathcal{ALCI}$ concepts, modulo the following (explicit) definitions: *(i)* $\forall R{:}C := \neg\exists R{:}\neg C$, *(ii)* $C \sqcup C' := \neg(\neg C \sqcap \neg C')$, *(iii)* $\bot := \neg\top$, and *(iv)* $\exists R := \exists R{:}\top$.

In a DL ontology $\mathcal{O}$, intensional knowledge is specified by means of a set of (concept inclusion) *assertions* of the form $C_l \sqsubseteq C_r$, stating inclusion (or IS-A) between the instances of the concept C_l on the *left* and those of the concept C_r on the *right*. In $\mathcal{ALCI}$, C_l and C_r may be arbitrary concepts, while fragments of $\mathcal{ALCI}$, such as *DL-Lite*$_\sqcap$ [6], can be obtained by suitably restricting the syntax for C_l and for C_r. A *database* (DB), expressing extensional knowledge, is a finite set $\mathcal{D}$ of unary and binary ground atoms of the form $A(c)$, $P(c, c')$. A *knowledge base* (KB) is a pair $\langle \mathcal{O}, \mathcal{D} \rangle$, where $\mathcal{O}$ is an ontology and $\mathcal{D}$ a DB.

As query language, we consider *conjunctive queries*, i.e., SELECT-PROJECT-JOIN SQL queries and *tree shaped conjunctive queries* (TCQs), which are those CQs that are built using only unary and binary relations and that are tree-isomorphic [12]. The *query answering* (QA) decision problem for (T)CQs and KBs is the (FOL) entailment problem stated as follows: given a KB $\langle \mathcal{O}, \mathcal{D} \rangle$, a sequence $\bar{c}$ of constants and a (T)CQ q, check whether there exists a grounded substitution σ of the free variables of q with $\bar{c}$ s.t. $\mathcal{O} \cup \mathcal{D} \models q\sigma$, i.e., whether $\mathcal{O} \cup \mathcal{D}$ (when seen as a FOL theory) entails the grounding of q (seen as a FOL open formula) by σ. We are interested in the *data complexity* of QA, namely, in its computational complexity when we consider $\mathcal{D}$ as the only input of the problem.

3 Expressing QA in Controlled English

Given an ontology language $\mathcal{L}$ and a query language $\mathcal{Q}$, to *express QA in controlled English* we: *(i)* Define a grammar $G_\mathcal{L}$ and a compositional translation $\tau(\cdot)$ for the corresponding controlled declarative fragment $L(G_\mathcal{L})$ s.t. $\tau(L(G_\mathcal{L})) = \mathcal{L}$. *(ii)* Define a grammar G_Q and a compositional translation $\tau'(\cdot)$ for the corresponding controlled interrogative fragment $L(G_\mathcal{Q})$ s.t. $\tau'(L(G_\mathcal{Q})) = \mathcal{Q}$. Following formal semantics terminology we call such ontology/query language expressions the *meaning representations* (MRs) of the CL utterance.

In [4, 15], we have shown how to express the DL *DL-Lite*$_\sqcap$ and TCQs, for which QA is in **L** [6], with the CLs Lite-English and GCQ-English, respectively. We want now

S → **NP VP**	**VP** → is a **Nom**	**VP** → is **TV** by **NP**	**NP** → **Det Nom**
VP → **TV NP**	**VP** → **IV**	**VP** → is **Neg TV** by **NP**	**NP** → **Pro Relp VP**
VP → is **Adj**	**VP** → is **Neg** a **Nom**	**Nom** → **Nom Relp VP**	**NP** → **Pro**
VP → does **Neg IV**	**VP** → **VP Crd VP**	**Nom** → **Nom Crd Nom**	**Nom** → **Adj Nom**
VP → is **Neg Adj**			**Nom** → **N**

$\tau(\textbf{VP}) := \tau(\textbf{NP})(\tau(\textbf{TV}))$	$\tau(\textbf{VP}) := \tau(\textbf{Crd})(\tau(\textbf{VP}))(\tau(\textbf{VP}))$	$\tau(\textbf{S}) := \tau(\textbf{NP})(\tau(\textbf{VP}))$
$\tau(\textbf{VP}) := \tau(\textbf{Neg})(\tau(\textbf{NP})(\tau(\textbf{TV})))$	$\tau(\textbf{VP}) := \tau(\textbf{Neg})(\tau(\textbf{Adj}))$	
$\tau(\textbf{VP}) := \tau(\textbf{Neg})(\tau(\textbf{Nom}))$	$\tau(\textbf{VP}) := \tau(\textbf{Neg})(\tau(\textbf{IV}))$	$\tau(\textbf{VP}) := \tau(\textbf{Adj})$
$\tau(\textbf{NP}) := \tau(\textbf{Pro})$	$\tau(\textbf{VP}) := \tau(\textbf{IV})$	$\tau(\textbf{VP}) := \tau(\textbf{Nom})$
$\tau(\textbf{NP}) := \tau(\textbf{Det})(\tau(\textbf{Nom}))$	$\tau(\textbf{NP}) := \tau(\textbf{Pro})(\tau(\textbf{Relp})(\tau(\textbf{VP})))$	$\tau(\textbf{Nom}) := \tau(\textbf{N})$
$\tau(\textbf{Nom}) := \tau(\textbf{Nom})(\tau(\textbf{Relp})(\tau(\textbf{VP})))$	$\tau(\textbf{Nom}) := \tau(\textbf{Crd})(\tau(\textbf{Nom}))(\tau(\textbf{Nom}))$	$\tau(\textbf{Nom}) := \tau(\textbf{Adj})(\tau(\textbf{Nom}))$

Pro → anybody	$\tau(\textbf{Pro}) := \lambda C.\lambda C'.C \sqsubseteq C'$	**Pro** → somebody	$\tau(\textbf{Pro}) := \lambda R.\exists R$
Pro → nobody	$\tau(\textbf{Pro}) := \lambda C.\lambda C'.C \sqsubseteq \neg C'$	**Pro** → nobody	$\tau(\textbf{Pro}) := \lambda R.\neg\exists R$
Crd → and	$\tau(\textbf{Crd}) := \lambda C.\lambda C'.C \sqcap C'$	**Crd** → or	$\tau(\textbf{Crd}) := \lambda C.\lambda C'.C \sqcup C'$
Relp → who	$\tau(\textbf{Relp}) := \lambda C.C$	**Relp** → who	$\tau(\textbf{Relp}) := \lambda C.\lambda C'.C{:}C'$
Neg → not	$\tau(\textbf{Neg}) := \lambda C.\neg C$	**Pro** → only	$\tau(\textbf{Pro}) := \lambda R.\lambda C.\forall R{:}C$
Pro → everybody	$\tau(\textbf{Pro}) := \lambda C.\top \sqsubseteq C$	**Pro** → nobody	$\tau(\textbf{Pro}) := \lambda C.C \sqsubseteq \bot$
Det → some	$\tau(\textbf{Det}) := \lambda R.\lambda C.\exists R{:}C$	**Det** → every	$\tau(\textbf{Det}) := \lambda C.\lambda C'.C \sqsubseteq C'$
Det → no	$\tau(\textbf{Det}) := \lambda C.\lambda C'.C \sqsubseteq \neg C'$		

Fig. 1. The grammar G_{DL} of DL-English.

to know which fragments of ACE-OWL are *(i) maximal* w.r.t. tractable data complexity (i.e., in **PTime**), and hence scale up with data, and *(ii) minimal* w.r.t. intractable data complexity (i.e., **coNP**-hard), and hence do not scale up.

Figure 1 introduces DL-English's grammar G_{DL}. Following DL conventions [3], we associate (and hence map) the non-recursive word categories **N**, **Adj** and **IV** to atomic concepts. Category **TV** is associated to role names. Recursive constituents, by contrast, are associated to arbitrary concepts. For reasons of simplicity and space, we disregard morphology and polarity issues. We also omit specifying the (open) class of content words. An example of a sentence recognized by DL-English (we spell out its MR underneath) is:

No man who runs some business that does not make some money is a businessman.

$$\mathit{Man} \sqcap \exists \mathit{run}{:}(\mathit{Business} \sqcap \neg(\exists \mathit{make}{:}\mathit{Money})) \sqsubseteq \neg \mathit{Businessman}$$

We now turn to the computational properties of each of the constructs *in isolation* of DL-English. We do it by essentially restricting the kind of *right* (i.e., C_r) and *left* (i.e., C_l) concepts we may express. All utterances comply with the sentence patterns

$$\text{"every } \alpha_l\ \alpha_r\text{"} \qquad \text{and} \qquad \text{"everybody who } \alpha_l\ \alpha_r\text{"}.$$

The constituents α_l and α_r map to, respectively, left and right concepts, while sentences map to IS-A assertions of the form $C_l \sqsubseteq C_r$. We consider in this paper only 8 out of all possible combinations obtained by allowing in C_l and C_r some subset of the DL constructs in the upper part of Figure 2, giving rise to the family $\{\text{IS-A}_i\}_{i\in[0,7]}$ of CLs shown in Figure 2. The basic kind of assertion they all express is IS-A among atomic concepts, viz., $A \sqsubseteq A'$, captured by IS-A_0.

The complexity result for IS-A_0 follow from the fact that Lite-English subsumes it. For all the other fragments, the complexity lower bounds follow from the results in [6]. The upper bounds for the CLs IS-A_i, for $i \in \{2, 3, 4\}$, follow from results by [10] for the DL $\mathcal{EL}$, which subsumes the DL assertions they express. Membership in **NL** for IS-A_1 follows by reducing QA over IS-A_1 KBs to linear datalog program evaluation, which is well-known to be in **NL** (w.r.t. data complexity) [1].

Concept C_f	**Constituent** α_f	**Grammar Rules**
A	**Nom**$_f$, **VP**$_f$	**VP**$_f$ $\rightarrow$ is a **Nom**$_f$ | **IV** | is **Adj** **Nom**$_f$ $\rightarrow$ **N**
$\exists P{:}A$	**TV** some **Nom**$_f$, **TV** somebody who **VP**$_f$	
$\exists P^{-}{:}A$	**TV** by some **Nom**$_f$, **TV** by somebody who **VP**$_f$	
$\forall P{:}A$	**TV** only **VP**$_f$, **TV** only who **VP**$_f$	
$\exists P$	**TV** something, **TV** somebody	$\emptyset$
$A_1 \sqcap \cdots \sqcap A_n$	**Adj Nom**$_f$, **Nom**$_f$ who **VP**$_f$ **Nom**$_f$ and **Nom**$_f$, **VP**$_f$ and **VP**$_f$	**VP**$_f$ $\rightarrow$ is a **Nom**$_f$ | **IV** | is **Adj** | **VP**$_f$ and **VP**$_f$ **Nom**$_f$ $\rightarrow$ **N** | **Adj Nom**$_f$ | **Nom**$_f$ and **Nom**$_f$
$A_1 \sqcup \cdots \sqcup A_n$	**VP**$_f$ or **VP**$_f$	**VP**$_f$ $\rightarrow$ is a **Nom**$_f$ | **IV** | is **Adj** | **VP**$_f$ and **VP**$_f$ **Nom**$_f$ $\rightarrow$ **N** | **Nom**$_f$ and **Nom**$_f$
$\neg A$	is not **Adj**, does not **IV**, is not a **Nom**$_f$	**Nom**$_f$ $\rightarrow$ **N**

Fragment	**Assertions**	**Sample Sentence(s)**	**Data Complexity**
IS-A$_0$	$A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n$	every businessman is a cunning man	in **L**
IS-A$_1$	$A \sqsubseteq \forall P{:}A$	every herbivorous eats only herbs	**NL**-complete
IS-A$_2$	$A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \forall P{:}(A_1 \sqcap \cdots \sqcap A_m)$	every Italian man drinks only strong coffee	**PTime**-complete
IS-A$_3$	$\exists P{:}A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n$ $\exists P^{-}{:}A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n$ $A \sqsubseteq \exists P$	anybody who murders some person is a heartless killer, anybody who is loved by some person is a happy person, every driver drives something	**PTime**-complete
IS-A$_4$	$A_1 \sqcap \cdots \sqcap A_n \sqsubseteq A_1 \sqcap \cdots \sqcap A_m$ $\exists P{:}(A_1 \sqcap \cdots \sqcap A_n) \sqsubseteq A_1 \sqcap \cdots \sqcap A_m$	every cruel man is a bad man, anybody who runs some bankrupt company is a bad businessman	**PTime**-complete
IS-A$_5$	$\forall P{:}A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n$	anybody who values only money is a greedy person	**coNP**-complete
IS-A$_6$	$A \sqsubseteq A_1 \sqcup \cdots \sqcup A_n$	every mammal is male or is female	**coNP**-complete
IS-A$_7$	$\neg A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n$	anybody who is not selfish is a reasonable person	**coNP**-complete

	Lite English	**EL-English**	**DL-English**	**ACE-OWL**
Data Complexity	in **L**	**PTime**-complete	**coNP**-complete	**coNP**-hard

Fig. 2. The $\{\text{IS-A}_i\}_{i\in[0,7]}$ fragments and DL-English, where $f \in \{l, r\}$ and IS-A$_i$, for all $i > 0$, contains also the assertions of IS-A$_0$.

We can now individuate the constructs of DL-English, and a fortiori of any CL expressing a **coNP**-hard ontology language such as $\mathcal{SHIF}$ (as does ACE-OWL) that negatively affect the scalability of CL interfaces to ODBASs, namely:

- "only" in subject position (**coNP**-hardness of IS-A$_5$),
- disjunction in predicate position (**coNP**-hardness of IS-A$_6$),
- negation in subject position (**coNP**-hardness of IS-A$_7$).

They also allow us to identify *maximal* CLs contained in DL-English (and a fortiori ACE-OWL) w.r.t. scalability (i.e., tractable data complexity). By merging the (tractable) fragments IS-A$_i$, for $i \leq 4$, we essentially express the $\mathcal{ELI}$ ontology language, with syntax (notice that the assertion $A \sqsubseteq \forall P{:}A'$ is equivalent to $\exists P^{-}{:}A \sqsubseteq A'$):

$$R \rightarrow P \mid P^{-} \qquad\qquad C \rightarrow \top \mid A \mid \exists R{:}C \mid C \sqcap C$$

That is, the DL where negation- and disjunction free existential concepts are allowed to arbitrarily nest on *both* sides of $\sqsubseteq$. $\mathcal{ELI}$ induces a **PTime**-complete fragment of DL-English, that we term EL-English, which captures most of the constraints and axioms of real-world large-scale biomedical ontologies such as GALEN or SNOWMED [2]. We can define EL-English top-down pretty easily by removing from G_{DL} the grammar rules for negation, disjunction, and universal quantification, and the negative function words. In such a CL arbitrary sentence subordination (and relatives), in combination

with, existential quantification and conjunction among **VP**s or **Nom**s is allowed. Universal quantification is highly controlled and negation and disjunction are ruled out.

4 Conclusions

We have studied the computational complexity of querying OBDASs in controlled English. Optimal (i.e., **L**) data complexity is attained with Lite-English (expressing *DL-Lite*$_{\sqcap}$) and GCQ-English (expressing TCQs) [4, 15]. Relaxing, however, the constraints put on negation, disjunction, universal determiners, and pronouns, until the components in sentence subjects and predicates become symmetrical, as in DL-English or more expressive CLs, yields **coNP**-hardness. In particular, data access in ODBASs through CL interfaces is intractable when coverage in declarations is extended to negation and universal quantification in subject **NP**s, or when we allow disjunction in predicate **VP**s. We also identify a maximal (scalable) CL, EL-English, which is in **PTime** w.r.t. data complexity.

Acknowledgements. We thank Raffaella Bernardi and Norbert Fuchs for discussions and criticism regarding earlier versions of this paper.

References

1. S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison Wesley, 1995.
2. F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of IJCAI 2005*.
3. F. Baader, D. Calvanese, D. Nardi, P. Patel-Schneider, and Deborah McGuinness. *The Description Logic Handbook*. Cambridge University Press, 2003.
4. R. Bernardi, D. Calvanese, and C. Thorne. Lite Natural Language. In *Proc. of the 7th Int. Workshop on Computational Semantics (IWCS-7)*, 2007.
5. A. Bernstein, E. Kaufman, A. Göhring, and C. Kiefer. Querying ontologies: A controlled English interface for end-users. In *Proc. of ISWC 2005*.
6. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Data complexity of query answering in description logics. In *Proc. of KR 2006*.
7. N. E. Fuchs and K. Kaljurand. Mapping Attempto Controlled English to OWL DL. In *Demos and Posters of the 3rd European Semantic Web Conf. (ESWC 2006)*.
8. I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. From $\mathcal{SHIQ}$ and RDF to OWL: The making of a web ontology language. *J. of Web Semantics*, 1(1):7–26, 2003.
9. K. Kaljurand and N. E. Fuchs. Verbalizing OWL in Attempto Controlled English. In *Proc. of OWLED 2007*.
10. A. Krisnadhi and C. Lutz. Data complexity in the $\mathcal{EL}$ family of description logics. In *Proc. of LPAR 2007*.
11. S. Mador-Haim, Y. Winter, and A. Braun. Controlled language for geographical information system queries. In *Proc. of Inference in Computational Semantics*, 2006.
12. M. Ortiz, D. Calvanese, and T. Eiter. Data complexity of query answering in expressive description logics via tableaux. *J. of Automated Reasoning*, 41(1):61–98, 2008.
13. I. Pratt and A. Third. More fragments of language. *Notre Dame J. of Formal Logic*, 2005.
14. M. Slavkovik. Deep analysis for an interactive question answering system. Master's thesis, KRDB Research Centre, Free University of Bozen-Bolzano, 2007.
15. C. Thorne. Expressing aggregate queries over DL-Lite ontologies with controlled English. In *Proc. of the ESSLLI 2008 Student Session*, 2008.
16. M. Y. Vardi. The complexity of relational query languages. In *Proc. of STOC'82*.