Managing Structured Data with Controlled English

Camilo Thorne

Supervisor: Dr. R. Bernardi
Co-supervisor: Pr. D. Calvanese

Dobbiaco, January 22, 2007

Outline

A. The problem

Compositionality

Why constraining English?

B. Constraining English

Controlled Languages

Fragments of English (Pratt & Third)

C. DL-Lite and Lite English

DL-Lite

Lite English

Expressive Power

D. Conclusions

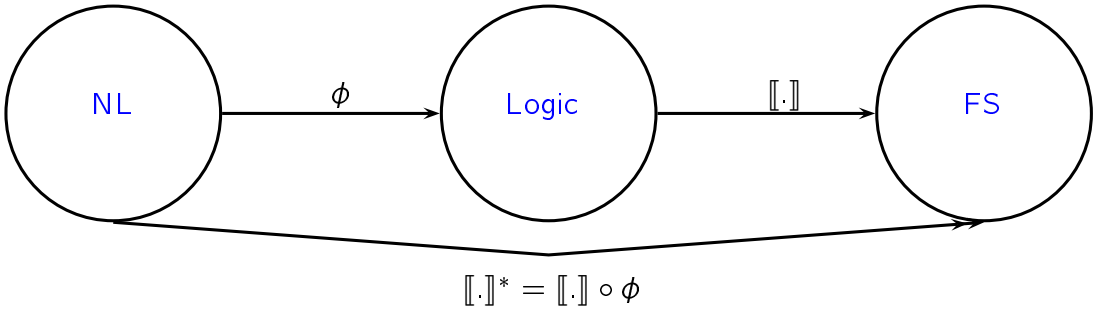
The Problem

- Natural language (NL) is not, *prima facie*, well-suited for querying and managing data or information stored in a relational database (DB) or a knowledge base (KB), let alone for engaging in knowledge representation (KR) tasks.
- On the other hand, formal query languages (like SQL, conjunctive queries, relational algebra or Datalog) and KR formalisms like description logics (DLs) have an unambigous syntax and semantics, together with a well-defined expressivity.
- It would be desirable to reach a trade-off between a restricted expressivity and the intuitive meaning of everyday speech for managing data.

Compositionality (1)

- A general framework for linking NL to logics is **formal semantics**.
- It provides the mathematical grounding for defining a **formal model-theoretic semantics** (FS) for NL. Moreover, it explains how to do this following the principle of **compositionality**: semantics = lexical semantics + phrase struture.
- The idea is to define a **compositional translation** ϕ (satisfying some conditions) from NL to a logic that, when composed with the interpretation function $[\![.]\!]$ mapping this logic to some FS, yields an interpretation $[\![.]\!]^*$ mapping NL to this FS.
- Compositional translations map NL utterances to logic formulas called **meaning representations** (MRs). Typically, first order logic (FOL) is enriched with **lambda calculus** expressions and syntactic composition is modelled by means of lambda application ([7]).

Compositionality (2)



Why Constraining English?

- We believe that using a subset of, say, English with a limited vocabulary and a restricted syntax can contribute in tackling the problem of managing data with NL by constraining the syntactic and semantic behaviour of English.
- This intuitively meaningful fragment will disallow ambiguity and **compositionally translate** into some logic – in our case, a KR logic and a formal query language.
- This approach has to be accompanied by a thorough study of the **expressive power** of the controlled language, since we want it to be as expressive as the query languages and the KR languages, but not more.
- This involves combining two heretofore different approaches to the issue: **controlled languages** and **fragments of English**.

Controlled Languages

- **Controlled languages** (CLs) are fragments of NL (say, of English), with a limited lexicon and set of grammar rules over which constraints are set to control the behaviour of components, both syntactically and semantically.
- Their main aim is to perform KR (and data management) tasks in NL (e.g. Attempto Controlled English, cf. [3, 6]).
- Their grammar (and ultimately their semantics) is engineered in such a way that every sentence has a unique parse and a MR (up to logical equivalence) belonging to some KR formalism or query language.

Fragments of English

- Pratt and Third in [9, 10, 11, 8] define a family of fragments of English and studies them w.r.t. expressive power.
- A **fragment of English** is some (intuitively meaningful) subset of English, defined by limiting to a strict minumum grammar rules, that compositionally tranlates into a fragment of FOL.
- COP, the minimal fragment, deals only with copula, negation, nouns (common and proper) and existential and universal quantification. COP+TV+DTV extends the coverage to transitive verbs and distransitive verbs. Other fragments are built by extending coverage to other words and syntactic constructs.
- **Expressive power** is defined in terms of that of the underlying meaning representation logic (a FOL fragment).

Building the Fragments

▷ **Syntactic constructs**:

COP	=	Copula, common and proper nouns, negation, universal and existential quantifiers.
TV	=	Transitive verbs (e.g. "reads").
DTV	=	Distransitive verbs (e.g., "gives").
REL	=	Relative pronouns (i.e., "who", "that", "which", etc.).
RA	=	Restricted (intrasentential) anaphora.
GA	=	Generalized anaphora.

▷ **COP**:

Every philosopher is a man	$\rightsquigarrow$	$\forall x[philosopher(x) \rightarrow man(x)]$
Socrates is a philosopher	$\rightsquigarrow$	$philosopher(Socrates)$
$\therefore$ Socrates is a man	$\rightsquigarrow$	$\therefore man(Socrates)$

Complexity of Fragments (Pratt & Third 2003)

Fragment	Decision class for satisfiability
COP	**P**
COP+TV+DTV	**P**
COP+REL	**NP-Complete**
COP+REL+TV	**EXPTIME-Complete**
COP+REL+TV+DTV	**NEXPTIME-Complete**
COP+REL+TV+RA	**NEXPTIME-Complete**
COP+REL+TV+GA	undecidable

DL-Lite and Lite English

- Description logics are decidable fragments of FOL specifically conceived for declaring, queriyng and reasoning over knowledge bases.
- Their formal properties are well known.
- They have the advantage that for some reasoning tasks they are more efficient than FOL (cf. [1]).
- What we need now is to choose the best-suited for querying – DL-Lite (cf. [4]).

DL-Lite (1)

- Let $A = \{A_i | i \in \mathbb{N}\}$ and $R = \{R_i | i \in \mathbb{N}\}$ be two countable sets of primitive **concept** and **role** symbols. Let $K = \{c_i | i \in \mathbb{N}\}$ be a set of **constants**.
- DL-Lite left (B) and right (C) hand side **concepts** are then defined as follows:

$$B ::= A | \exists R | \exists R^- | B \sqcap B.$$

$$C ::= B | \neg A | \neg \exists R | \neg \exists R^- | \exists R : C | \exists R^- : C.$$

- And DL-Lite ABox (A) and TBox (T) **assertions** as follows:

$$A ::= B(K) | R(K, K) | A(K).$$

$$T ::= B \sqsubseteq C.$$

- A DL-Lite **knowledge base** (KB) is a set of ABox and TBox assertions.

DL-Lite (2)

- DL-Lite corresponds to a tractable fragment of FOL.
- This is because operators such as, for instance, negation and the universal and existential quantifiers cannot be used freely.
- TBox satisfiability is in **P** and query answering is **LOGSPACE** in data complexity (i.e., on the number of constants in the ABox), when we consider simple conjunctive queries (CQs) over the KB.
- Using DL-Lite as a meaning representation formalism means that reasoning tasks can be carried out efficiently by means of tools such as QONTO (cf. Calvanese *et al.* [5]).

Conjunctive Queries

- A **simple conjunctive query** (CQ) is an expression of the form:

$$q(\vec{x}) \leftarrow r_1(\vec{y}_1), \ldots, r_n(\vec{y}_n).$$

- The r_i's, for $i \in [1, n]$, called **relations**, are DL-Lite basic concept and relation symbols. $\vec{x}$ is a possibly empty finite sequence of **distinguished variables**. The $\vec{y}_i$'s, for $i \in [1, n]$, are finite sequences of variables. The variables of $\vec{x}$ must occur among the r_i's, for $i \in [1, n]$ (**safeness**).
- CQs and KBs share the same alphabet of concepts, constants and relations.

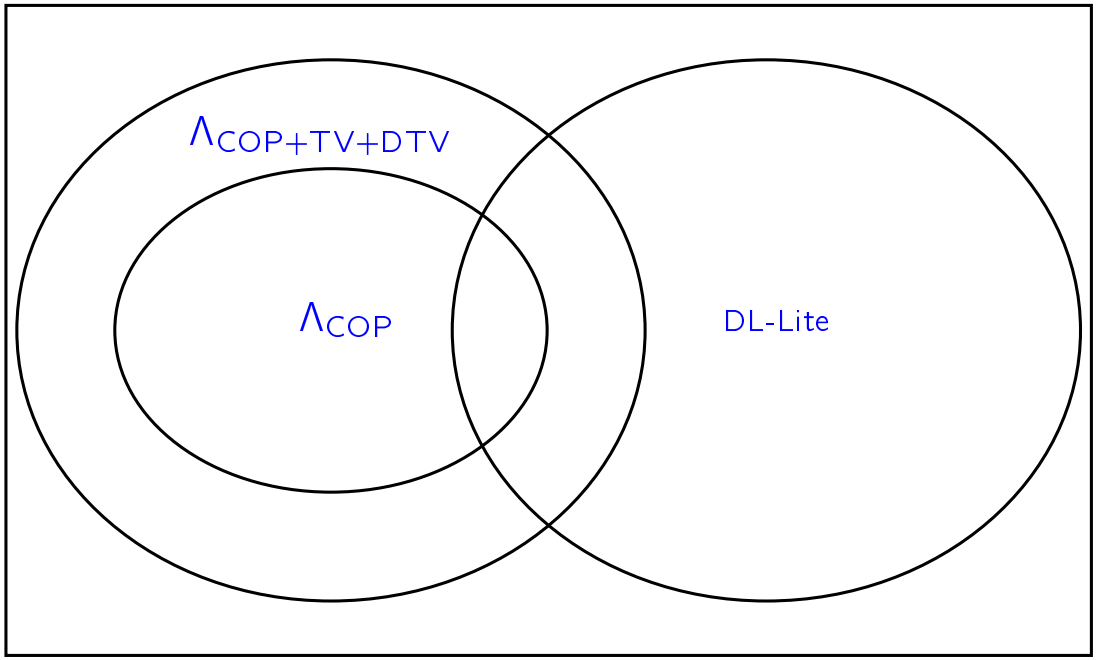
Lite English (1)

- Lite English is the controlled language we are developing to manage data following the approach set above: by constraining English so that expressive power coincides exactly with the one required by the data management tasks involved.
- It is divided in two fragments: **(i)** the declarative fragment and **(ii)** the fragment of questions.
- The former should compositionally translate into DL-Lite Abox and TBox assertions, the second to CQs.
- This implies (trivially) that Lite English and DL-Lite share the same expressive power.

Lite English (2)

- Lite English must allow (in its declarative fragment), for instance, negation only over nouns, adjectives and verbs in predicate position (but not over VP's or any other recursive component), since DL-Lite only allows for negation on right hand side concepts of only certain kinds – a very restricted use.
- It allows for a controlled use of relative pronouns.
- This is achieved as with other CLs by constraining the behaviour of components (**subcategorization**).
- This characteristic, subcategorization, is independant of the grammar formalism. We have tried a categorial grammar (CG) (cf. Bernardi *et al.* [2]) and a unification phrase structure grammar (UPSG). (cf. [12]).

Expressive Power of DL-Lite



Expressive Power of Lite English

- Since Lite English inherits the expressiveness of DL-Lite itself, its satisfiability problem is in **P**. Likewise, its querying problem falls under **LOGSPACE**.
- Moreover, it allows for a restricted use of relative clauses without yielding a state explosion.
- This is relevant, since the addition of relatives to COP or COP+TV+DTV yields untractability.
- This comes from the fact that relative pronouns are constrained in their use both in subject and predicate position, thus precluding full-blown gap-filler dependencies.

Conclusions

- We have defined a CL, Lite English, based on the KR logic DL-Lite sharing its expressive power, hence tractable.
- It is therefore suitable for representing and managing data in a way similar to DL-Lite, but in a way understandable for a casual end-user.
- This is good news, since in this way we have bridged the gap between CLs and the expressivity required by the tasks they are conceived.
- This is just a first step. Now, we have to deal with the problem of increasing coverage without going beyond those tight expressivity bounds.

References

[1] F. Baader, D. Calvanese, D. Nardi, P. Patel-Schneider, and D. McGuinness. *The Description Logic Handbook*. Cambridge University Press, 2003.

[2] Raffaella Bernardi, Diego Calvanese, and Camilo Thorne. Lite Natural Language. In *Proceedings of the 7th International Workshop on Computational Semantics (IWCS-7)*, 2007.

[3] Abraham Bernstein, Esther Kaufman, and Christoph Kiefer Anne Göring. Querying Ontologies: A Controlled English Interface for End-Users. www.ifi.unizh.ch/ddis/staff/goehring, 2003.

[4] Diego Calvanese, Giusseppe de Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. DL-Lite: Tractable Description Logics for Ontologies. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 2005)*, 2005.

[5] Diego Calvanese, Giusseppe de Giacomo, Domenico Lembo, Maurizio Lenzerini, and Ric-

cardo Rosati. Efficiently Managing Data Intensive Ontologies. In *Proceedings of the 2nd Italian Semantic Web Workshop: Semantic Web Applications and Perspectives (SWAP 2005)*, 2005.

[6] Norbert E. Fuchs, Kaarel Kaljurand, and Gerold Schneider. Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces. http://www.ifi.unizh.ch/attempto/publications, 2005.

[7] L. T. F. Gamut. *Logic, Language and Meaning (2 vols.)*. University of Chicago Press, 1991.

[8] Ian Pratt. On the Semantic Complexity of some Fragments of English. Technical report, Department of Computer Science – University of Manchester, 2001.

[9] Ian Pratt. A Two-Variable Fragment of English. *Journal of Logic, Language and Information*, (12), 2003.

[10] Ian Pratt. Fragments of Language. *Journal of Logic, Language and Information*, (13), 2004.

[11] Ian Pratt and Allan Third. More Fragments of Language. *Notre Dame Journal of Formal*

Logic, 2005.

[12] Camilo Thorne. Controlled English for DL-Lite. Technical report, KRDB Group – Free University of Bolzano, 2007.

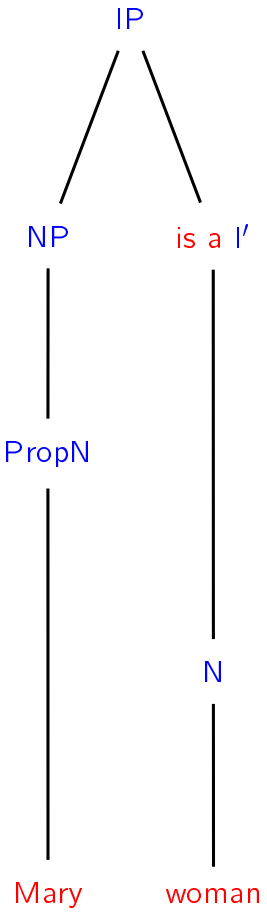
An Example - COP (1)

Syntax Rules	**MR** $(= \phi)$
IP $\rightarrow$ **NP I′**	$(\phi(\textbf{NP}))\phi(\textbf{I}') \rhd_\beta \phi(\textbf{IP})$
I′ $\rightarrow$ is a **N**	$\phi(\textbf{I}) = \phi(\textbf{N})$
I′ $\rightarrow$ is not a **N**	$\phi(\textbf{I}) = \neg\phi(\textbf{N})$
NP $\rightarrow$ **PropN**	$\phi(\textbf{NP}) = \phi(\textbf{PropN})$
NP $\rightarrow$ **Det N**	$(\phi(\textbf{Det}))\phi(\textbf{N}) \rhd_\beta \phi(\textbf{NP})$

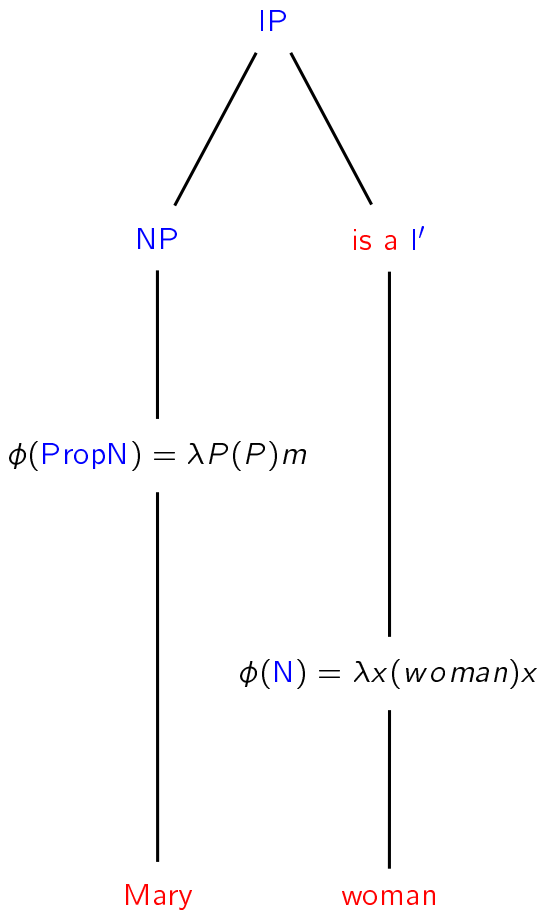
An Example - COP (2)

Lexicon	**MR** $(= \phi)$
N $\rightarrow$ woman	$\phi(\mathbf{N}) = \lambda x(woman)x$
N $\rightarrow$ man	$\phi(\mathbf{N}) = \lambda x(man)x$
N $\rightarrow$ human	$\phi(\mathbf{N}) = \lambda y(human)x$
PropN $\rightarrow$ Mary	$\phi(\mathbf{PropN}) = \lambda P(P)m$
Det $\rightarrow$ every	$\phi(\mathbf{Det}) = \lambda P \lambda Q \forall x[(P)x \rightarrow (Q)x]$
Det $\rightarrow$ no	$\phi(\mathbf{Det}) = \lambda P \lambda Q \forall x[(P)x \rightarrow \neg(Q)x]$
Det $\rightarrow$ some	$\phi(\mathbf{Det}) = \lambda P \lambda Q \exists x[(P)x \wedge (Q)x]$

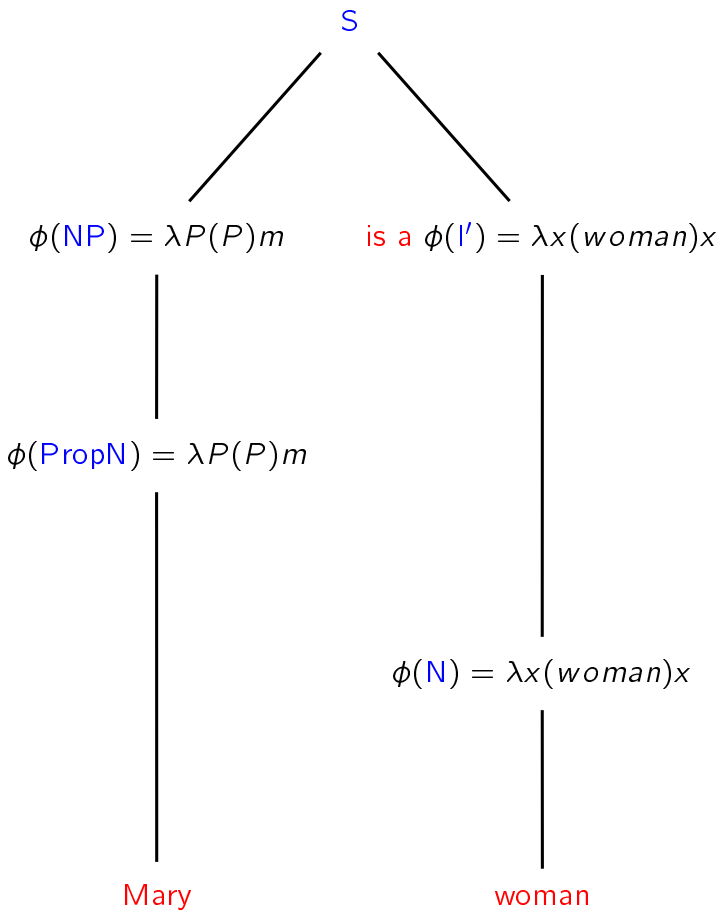
An Example - COP (3)



An Example - COP (4)



An Example - COP (5)



An Example - COP (6)