Open theories and abduction for context and accommodation

Henning Christiansen

Roskilde University, Computer Science Dept.,
P.O.Box 260, DK-4000 Roskilde, Denmark
E-mail: `henning@ruc.dk`

Abstract. A model is proposed for the representation of context by means of open logic theories parameterized by metavariables covered by constraints. The model is formalized and implemented using constraint logic and metalogic programming notions. Accommodation, understood as the process of extracting presuppositions embedded in a discourse or, in general, extracting the contents embedded in a stream of observations, fits naturally into this model as abduction. A possible worlds' semantics is given similar to a proposal given by Stalnaker.

1 Introduction

Context in linguistics and artificial intelligence usually refers to the amount of common knowledge shared among the participants in a discourse or to the knowledge currently held by an agent attending a discourse. As the discourse proceeds, more and more knowledge is gained which means that the context becomes more and more specific as the result of an accommodation process. This paper describes an abstract model and implementation of context and accommodation which can be seen as a formalization of Stalnaker's informal characterization summarized in [13], which inspired to this more systematic presentation based on earlier *ad hoc* experiments with simplified natural language analysis by means of abduction in a metalogic setting [4, 5]. For a full version of this paper with more details, examples and comprehensive references, see [9].

2 The basic model

In order to model the dynamic evolution of context, i.e., accommodation, we consider *open logic theories* useful. A theory being open means, at the semantic level, that several interpretations (i.e., worlds in a Kripke-style semantics) are possible, namely all those that are compatible with the knowledge gained from the discourse so far, perhaps restricted within a set of "reasonable" interpretations (worlds). We distinguish between an *object language* in which theories about states of affairs can be formulated and a *metalanguage* in which properties about such theories are expressed. Some object sentences are *closed theories* and we assume a set of possible *worlds* and a *semantic function* w mapping each closed theory T into a world $w(T)$.[1] An entailment relation $w \models s$ is assumed with the intended meaning that object sentence s holds in w; for a set of worlds W we write $W \models s$ iff $w \models s$ for all $w \in W$. A *naming relation* is assumed, giving for each element o of the object language a metalevel term $\lceil o \rceil$ which serves as a name for o. An *open theory* arises when parts of the name for a theory are left out, indicating their positions by metavariables. Metavariables inside name terms are indicated by reverse brackets so, e.g., $\lceil p(Y) \leftarrow \lfloor B \rfloor \rceil$ is a scheme for names of object clauses whose head is $p(Y)$ and whose body is unknown, indicated by the metavariable B. A metalevel term which can be instantiated to the name of a theory is said to be of *theory type*.

A *context* is a pair $K = \langle T, C \rangle$ where T is a metalevel term of theory type and C a metalevel formula; T is called the *theory part* and C the *constraint part* of K. The intended meaning is that a given context captures all those possible worlds contained in the open theory represented by T, however, only those permitted by C. We make this precise by the following definition: The *context set* for a context $\langle T, C \rangle$ is the set of worlds $W(\langle T, C \rangle)$ defined as

$$W(\langle T, C \rangle) = \{ w(t) \mid \langle \lceil t \rceil, C' \rangle \text{ is a ground instance of } \langle T, C \rangle \text{ in which } C' \text{ is true} \}$$

Open theories fit well with accommodation understood as specialization and with abduction applied as the means for specialization; abduction means to reason backwards using rules already present in the context, as background knowledge or perhaps learned during the discourse, in order to identify contextual facts that can explain the observed actions (utterances, sensor signals, etc.). We assume a class *Action* of object formulas (typically facts) called *actions*; a sequence of actions is called a *discourse*. A *specialization* of a context $\langle T, C \rangle$ is a new context $\langle T, C \wedge S \rangle \sigma$ where σ is a metalevel substitution and S a metalevel formula. An *accommodation function* is a function

$$\text{accomodate} : \mathit{Context} \times \mathit{Action} \rightarrow \mathit{Context}$$

where $\text{accommodate}(K, A)$ is a specialization of K; *Context* refers to the set of all contexts. An accommodation function is extended for sequences of actions in the natural way, and if it reaches a context $\top$ with $W(\top) = \emptyset$, it is said to fail. Some nonmonotonic aspects can be accounted for by allowing accommodation functions to be nondeterministic in the sense that they may produce more than one resulting new context; this fits with the backtracking provided by an implementation in logic programming. We require that an accommodation function satisfies the property that, for all contexts K and actions A, $W(\text{accomodate}(K, A)) \models A$, a definition that includes abduction and induction as special cases. Accommodation starts from an *initial context* which may include "background knowledge"

[1] There are no open theories represented at the object level, openness is obtained by parameterization at the metalevel. We chose the closed-theory–single-world formulation in order to stress that the interesting degrees of freedom are expressed at the level of open theories. The framework can easily be extended for, say, disjunctive object theories, by having $w(t)$ to be a (finite) set of worlds.

as well as metaknowledge to guide the accommodation function corresponding to, e.g., integrity constraints in database update, "bias" in inductive logic programming, or the "abducibles" in abduction.

3 Representation in metalogic programming

This model can be embedded in the DEMO system [5, 8] which is a constraint-based metaprogramming systems built on top of Prolog. The object language is that of positive Horn clauses and we let $w(P)$ be the least Herbrand model of P for any such program. The most important metalevel predicates are two proof predicates demo($\lceil P \rceil$, $\lceil Q \rceil$) and fails($\lceil P \rceil$, $\lceil Q \rceil$) with the meaning that object query Q succeeds, resp. fails, in the object program P. The interesting property of this system is the *reversibility* of demo: It works correctly also when the program argument contains metavariables standing for unknown parts of the object program. In this case, the execution of demo may generate object programs which make the object query provable. The fails predicate works basically as a metalevel version of negation-as-failure that delays subcomputations for the "missing parts" of the object program, thus providing an incremental evaluation of integrity constraints [6].

We can sketch the implementation of accommodation by the following metalevel query which captures the accommodation of a single action.

$$\mathrm{abd(A)},\ \mathrm{fails}(\lceil CT \,\&\, IC \,\&\, \lfloor \mathrm{A} \rfloor \rceil,\ \lceil \perp \rceil),\ \mathrm{demo}(\lceil CT \,\&\, \lfloor \mathrm{A} \rfloor \rceil,\ \lceil Action \rceil)$$

The 'abd' predicate defines object programs consisting of abducible atoms, CT is the know part of the theory in the current context. Integrity constraints are specified by an object program IC defining the predicate $\perp$. The metavariable A is the "opening" in the current context which will be partly instantiated during the execution of this query as to reach a new context in which the observed *Action* can be explained. See [5, 7, 9] for examples.

4 Related work

Compared with earlier models of context, e.g., [1, 12] and successors, this is a simplification in the sense that the general mechanism of abduction only refers to rules appearing in a description of the current domain so that an additional and orthogonal layer of so-called lifting rules or specialized modal operators becomes unnecessary. The model proposed here can host many existing knowledge representation formalisms, e.g., conceptual structures [14], terminological logic [2], and various other apparata biased towards the syntax and semantics of natural language. The fact that the framework is embedded in logic programming provides a direct interface to syntax analysis methods of which numerous have been described in the literature.

The use of abduction for natural language analysis is not new, e.g., [10]; our contribution is mainly at the semantic level, giving a new formulation in a metalogic setting which integrates abduction with open theories and possible worlds' semantics. Open logic theories have been studied in different shapes, e.g. [11, 15]. Our version differs from the mentioned by a more "fine-grained" parameterization obtained by metavariables that may stand in arbitrary positions in the expression naming the theory and controlled by arbitrary metalogic constraints.

Acknowledgment: This research is supported in part by the OntoQuery and DART projects funded by the Danish Research Councils.

References

1. Attardi, G., Simi, M., Metalanguage and reasoning across viewpoints. *Proc. ECAI94: Advances in Artificial Intelligence* O'Shea, T.O., (ed.), pp. 315–324. Elsevier Science Publishers, 1984.
2. Brachman, R.J., Levesques, H.J., Competence in knowledge representation. *Proceedings of the National Conference on Artificial Intelligence. Pittsburgh, PA, August 18–20, 1982*, Walz, D.L., AAAI Press pp. 189–192, 1982.
3. Buvač, S., Quantificational Logic of Context. *Proc. AAAI 96 & IAAI 96, Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference*, pp. 600–606, AAAI Press/MIT Press, 1996.
4. Christiansen, H., Why should grammars not adapt themselves to context and discourse? *4th International Pragmatics Conference, Kobe, Japan, July 23–30 1993*, (Abstract collection), International Pragmatics Association p. 23, 1993. (Extended abstract: http://www.dat.ruc.dk/~henning/IPRA93.ps).
5. Christiansen, H., Automated reasoning with a constraint-based metainterpreter, *Journal of Logic Programming*, vol. 37, pp. 213–253, 1998.
6. Christiansen, H., Lazy negation-as-failure for incremental integrity checking, *To appear*, 1999.
7. Christiansen, H., Abduction and induction combined in a metalogic framework. *Abductive and Inductive Reasoning: Essays on their Relation and Integration*, Flach. P., Kakas, A., (eds.), To appear, 1999.
8. Christiansen, H., Martinenghi, D., *The DemoII system.* Source code for implemented system, example files, and manuals available by World Wide Web, http://www.dat.ruc.dk/software/index.htm. Released 1998.
9. Christiansen, H., Open theories and abduction for context and accommodation. Full version of present paper. *To appear*, 1999 (currently available from http://www.dat.ruc.dk/~henning).
10. Hobbs, J.R., Stickel, M.E., Appelt D.E., and Martin, P., Interpretation as abduction. *Artificial Intelligence* 63, pp. 69-142, 1993.
11. Mancarella, P., Pedreschi, D., An algebra of logic programs. *Proc. Fifth International Conference and Symposium of Logic Programming*, pp. 1006–1023, MIT Press, 1988.
12. McCarthy, J., Generality in artificial intelligence. *Communications of the ACM*, vol. 30, pp. 1030–1035, 1987.
13. Stalnaker, R., On the representation of context. *Journal of Logic, Language, and Information*, vol. 7, pp. 3–19, 1998.
14. Sowa, J.F., *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, 1984.
15. Verbaeten, S., Denecker, M., De Schreye, D., Compositionality of Normal Open Logic Programs, Logic Programming, Proceedings of the 1997 International Symposium, Maluszynski, J., (ed.), pp. 371–385, MIT Press, 1997.