Inference in Computational Semantics

CHRISTOF MONZ, *Institute for Logic, Language and Computation, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands. E-mail: christof@wins.uva.nl*

MAARTEN DE RIJKE, *Institute for Logic, Language and Computation, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands. E-mail: mdr@wins.uva.nl*

The first workshop on Inference in Computational Semantics (ICoS-1) was held at the Institute for Logic, Language and Computation of the University of Amsterdam, on August 15, 1999. It was aimed at bringing together researchers from scientific disciplines to discuss approaches and applications of inference in natural language semantics, ranging from theoretical to implementational issues. In particular, ICoS-1 featured three invited presentations, six contributed talks, and six system demonstrations showing how inference tools are used in natural language applications. The variety of the program was also mirrored by the spectrum of contributing authors, which included researchers from Computational Linguistics, Artificial Intelligence, Computer Science, and Logic.

This special issue is derived from ICoS-1. In it we bring together nearly all of the papers presented at the workshop, as well as short descriptions of the systems that were demonstrated at ICoS-1. Before giving a brief description of the contributions, we discuss some generalities.

1 What is Computational Semantics?

Semantics is defined as the study of meaning expressed by elements of a language or combinations thereof. Natural language utterances are not just noises or scribbles: they are used to convey information, they are linked with kinds of events, with states of mind, etc. Natural language semantics, as a theory of meaning, determines what the meanings of words of the language are and how to semantically combine elements of a language to build up complex meanings. These meanings are most often represented as formulas in a logical language.

Computational semantics investigates the computational properties that formal semantic theories need to enjoy to be applicable to real-world problems. Most approaches in formal semantics are evaluated with respect to criteria like *expressiveness*, *explanatory adequacy*, and *generality*, whereas typical criteria for evaluating approaches in computational semantics include *coverage*, *robustness*, *efficiency*, and *user-friendliness*. *Coverage* measures the amount of phenomena that can described by a semantic theory. Although coverage should be regarded as a criterion for evaluating formal semantic theories, the impact of a good or bad coverage is best noticed when considering practical applications. Experience in computational linguistics, e.g. parsing, has shown that the coverage of sophisticated formalisms is generally rather small, which might be due to the fact that most of the research focussed on 'interesting', non-trivial phenomena, neglecting more frequent and less appealing phenomena [5].

Robustness is strongly related to coverage. An implementation is considered to be robust if it does not abrupt the execution in case the input is not covered by the theory. This criterion is becoming increasingly important because the amounts of systems are expected to be able to deal with are heavily growing.

Efficiency describes the performance of a system in terms of time and resource behavior. What is considered to be efficient depends of course on the application. A system categorizing a large collection of documents within 24 hours can be called efficient whereas a natural language dialogue system answering a user's question after five minutes cannot be. The need of efficiency requires careful thinking about the algorithmic encoding of formal semantic theories.

User-friendliness is a more general criterion describing the usefulness of computational semantic system for real applications. At the moment the number of applications that involve computational semantic technology is rather limited, but it is a generally held belief that the potential for those technologies is very promising, cf. [1, 7]. Systems that are currently available, such as DORIS or TRINDI, are mainly designed for research purposes and are not accessible by users that do not have any background in formal semantic theories like Discourse Representation Theory or Montague Semantics, which of course cannot be assumed if a system is intended to be used by a larger public.

Which of those criteria is considered to be the more important depends on the particular area of application. See [4, 6, 8] for more general criteria for evaluating natural language processing systems.

2 The Role of Inference in Computational Semantics

As mentioned above, logic is the most prevalent way of representing the semantics of natural language. Up to now, most work within computational semantics has focused on representational aspects, and deductive reasoning is still in its infancy, see [10]. In contrast, one of the concluding slogans of the FraCaS project on Frameworks for Computational Semantics is that '[t]here can be no semantics without logic' [3]. We take this to mean that formalisms for semantic representation should be developed hand-in-hand with inference methods for performing reasoning tasks with representations and algorithms for representation construction.

Clearly, to be usable in the first place, representation formalisms need to come equipped with construction methods, and this explains the need for algorithmic tools. But what about the need for inference methods? At least types of three reasons can be identified. For *cognitive* purposes one may want to test the truth conditions of a representation against (a model of) speakers' intuitions — this amounts to a model checking or theorem proving task. Also, the whole issue of what it is to understand a discourse may be phrased as a model generation task. *Computationally*, we need various reasoning tasks and AI-heuristics to help resolve quantifier scope ambiguity, or to resolve anaphoric relations in information extraction and natural language queries. And last, but not least, the very construction of semantic representations may require inference tools to be used in checking for consistency and informativity. At the end of the day, the main purpose of a semantic representation is that we can *do* something with it, both algorithmically and in terms of inference tasks.

The present times are exciting ones for anyone with an interest in inference for natural language semantics. On the one hand, there is work in semantics that has little or no attention for inferential aspects. This is certainly the case for a lot of work in dynamic semantics and underspecified representation, and in the recent *Handbook of Logic and Language* [9] inferential methods for semantic representations are largely absent, despite the fact that a substantial part of the book is devoted to representational matters. At the same time, there is a growing body of work aimed at developing inference methods and tools for natural language semantics, fed by a growing realization that these are 'the heart of the enterprise' [2, page viii]. This is manifested not only by various research initiatives (see below), but also by the fact that a number of textbooks and monographs on natural language semantics and its inferential and algorithmic aspects are in preparation [2, 3], and by a recent initiative to set up a special interest group on Computational Semantics (see `http://www.coli.uni-sb.de/~patrick/SIGICS.html` for details).

These developments are being re-enforced by the fact that traditional inference tools (such as theorem provers and model builders) are reaching new levels of sophistication and are now widely and easily available. A wide variety of new tools (statistical and probabilistic methods, ideas from the machine learning community) are likely to be increasingly applied in computational semantics. Most importantly of all, computational semantics seems to have reached the stage where the exploration and development of inference is one of its most pressing tasks — and there is a lot of interesting new work which takes inferential issues seriously.

3 The Contributions to ICoS-1

For reasons beyond our control, contributions based on the invited presentations by Johan Bos and Stephen Pulman could not be included in this issue. In his presentation, Bos discussed the role of (first-order) inference in contemporary and future semantic analysis within natural language processing. The discussion was centered around the use of automated reasoning in the experimental human-machine dialogue system Midas, in which inference tasks are farmed out to many different provers simultaneously.

Pulman, described a formalism and implementation for the interpretation and generation of sentences containing context dependent constructs like determiners, pronouns, focus, and ellipsis. In his proposal, a variant of 'quasi-logical form' is used as an underspecified meaning representation, related to 'resolved logical forms' via 'conditional equivalences'. These equivalences define the interpretation of contextually dependent constructs with respect to a given context. Higher-order unification and abduction are used in relating expressions to contexts.

We're very glad that a contribution by the third invited speaker, Matthew Stone, did make it to this special issue.

Research Papers

While a variety of issues were discussed at ICoS-1, some clear main themes could be discerned. Among these are the use of model generation as opposed to other reasoning tasks such as consequence checking. The following papers address this issue.

Models and Discourse Models: Ramsay and Seville The main message of this paper is that using inference engines for constructing models, rather than for drawing out consequences is a promising way forward. The authors argue that to assimilate an utterance, one's task is to construct a model that supports the content of the utterance itself, one's background encyclopaedic knowledge of the world, and the content of the discourse so far. After one has taken an utterance on board, one no longer has access to the utterance itself: all that is left is the picture of the world ('the model') that has been created.

Abducing Coreference by Model Construction: Baumgartner and Kühn. This paper applies model generation methods to resolve anaphora. The authors argue that the resolution of anaphoric expressions in an utterance is essentially an abductive task, and give a semantic representation for utterances containing anaphora that enables them to compute possible antecedents by abductive inference. This abductive inference is applied to the resolution of anaphoric expressions by using a hyper-tableaux based general model constructing framework.

A Natural Language Front-End to Model Generation: Fuchs, Schwertel, and Torge. A subset of English, called *Attempto Controlled English (ACE)*, is described that allows domain specialists to express problems in the language of their application domain. This controlled fragment can be unambiguously translated into first-order logic, and thus can replace first-order logic as a formal notation. ACE can be used as a front-end to a model generation method for verification of program specifications.

Interpreting Definites using Model Generation: Gardent and Konrad. In this paper, model generation is used to provide a procedural interpretation for semantic theories of natural language. The approach is illustrated by a description of how singular definite descriptions can be interpreted using the higher-order model generator.

In addition to model generation, a second major theme shared by many presentations at ICoS-1 was the practical use of semantics for content-intensive tasks such as information retrieval, knowledge management, and dialogue systems.

Towards a Computational Account of Knowledge, Action and Inference in Instructions: Stone. The author considers abstract instructions, which provide indirect descriptions of actions in cases when a speaker has key information that a hearer can use to identify the right action to perform, but the speaker alone cannot identify that action. The generation of abstract instructions requires a system to assess whether an instruction provides sufficient information for the user to draw appropriate inferences about action from it. Stone sketches a framework for specifying, computing, and accessing those assessments in natural language generation.

Reason Maintenance in a Hybrid Reasoning System: Kaplan. This paper discusses the conflict between two knowledge representation techniques: reason maintenance and special-purpose representations. Reason maintenance keeps track of a system's justifications for holding certain beliefs. Special-purpose representations are often used by inference systems to gain efficiency.

An Inference-Based Approach to the Interpretation of Discourse: Ludwig, Görz, and Nieman. This paper presents a decidable version of Discourse Representation Theory based on Description Logics. It shows how terminological reasoning can be integrated into parsing, while reasoning about the extensional meaning of utterances serves for realizing mixed-initiative dialogues.

Inference through Alternative-Set Semantics: Bierner and Webber. Alternative set semantics can be used to analyze a class of lexical items frequently found in natural language requests to search engines and databases, including *other (than), such (as),* and *besides*. The semantic representation of those requests is couched in combinatory categorial grammar. Most approaches to information retrieval simply neglect those words, but the authors show how an alternative-set semantics can help to improve retrieval.

System Descriptions

While quite a few of the paper presentations at ICoS-1 were accompanied by a system demonstration, there was also a small number of 'stand alone' demonstrations.

Natural Language Specifications for Hardware Verification: Holt, Klein and Grove. The authors describe a system that allows the formal verification of digital circuits using specifications expressed in English, using the model checking program SMV. The complete system demonstrates a language engineering application allied to a specialised reasoning engine.

Action Formalisms in Language Understanding: Otero and Trinidad. The authors present a natural language system that can construct an action theory that represents the meaning of a text in restricted English. They show how nonmonotonic temporal inference is needed for consistency checking and anaphora resolution in evolving domains, where conditions may change along time.

Classifying Scope Ambiguities: Gabsdil and Striegnitz. This paper describes the architecture and implementation of a system which compares and sorts semantic representations of natural language input with respect to equivalence of logical content and context change potential. By using automated theorem proving we compute a graph-like structure which represents the relationships that hold between different readings of a given sentence.

4 For the Record

We conclude this editorial introduction by thanking everyone who helped us to turn ICoS-1 into a success.

Program Committee

The program committee for ICoS-1 consisted of James Allen (Rochester), Patrick Blackburn (Saarbrücken), Denys Duchier (Saarbrücken), Jan van Eijck (Amsterdam/Utrecht), Claire Gardent (Saarbrücken), Jacques Jayez (EHESS/CELITH), Aravind Joshi (Philadelphia), Michael Kohlhase (Saarbrücken), Alex Lascarides (Edinburgh), Christof Monz (Amsterdam), Reinhard Muskens (Tilburg), Manfred Pinkal (Saarbrücken), Maarten de Rijke (Amsterdam), Len Schubert (Rochester), and Henk Zeevat (Amsterdam).

Out of 16 submissions, the program committee selected 7 papers for presentation at the workshop. In addition, Johan Bos, Stephen Pulman, and Matthew Stone were invited to address the meeting.

Sponsors

ICoS-1 was sponsored by the Computational Logic group at the Faculty of Science of the University of Amsterdam; the Institute for Logic, Language and Computation (ILLC); Chinaski WorldWide; the Netherlands Organization for Scientific Research (NWO), and the Deutsche Forschungsgemeinschaft (DFG). We gratefully acknowledge their generous support.

Acknowledgments

We would like to thank Rafael Accorsi and Marco de Vries for practical assistance before and during ICoS-1. We are also grateful to the editors of the *Journal of Language and Computation* for their enthusiasm and guidance.

References

[1] J.F. Allen. *Natural Language Understanding*. Benjamin Cummings, second edition, 1994.

[2] P. Blackburn and J. Bos. *Representation and Inference for Natural Language*. Available from `http://www.coli.uni-sb.de/~bos/comsem/`, 1998.

[3] D. Crouch, C. Fox, J. van Genabith, and S. Glasbey, editors. *The FraCaS Book*. Unpublished manuscript, 1998.

[4] EAGLES. Evaluation of natural language processing systems. Technical Report EAG-EWG-PR.2, Expert Adisory Group on Language Engineering Standards, 1995.

[5] A. Franz. *Automatic Ambiguity Resolution in Natural Language Processing: An Empirical Approach*. Lecture Notes in Artificial Intelligence 1171. Springer, 1996.

[6] M King. Evaluating natural language processing systems. *Communications of the ACM*, 39(1):73–79, 1996.

[7] S.J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

[8] K. Sparck Jones. *Evaluating Natural Language Processing Systems*. Lecture Notes in Artificial Intelligence 1083. Springer, 1996.

[9] J. van Benthem and A. ter Meulen, editors. *Handbook of Logic and Linguistics*. Elsevier Science Press, 1997.

[10] K. van Deemter and S. Peters, editors. *Semantic Ambiguity and Underspecification*. CSLI Publications, 1996.

Received February 29, 2000