

Book Review

Roland Hausser. *A Computational Model of Natural Language Communication*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN-13 978-3-540-35476-5. ii+365 pages. doi:10.1017/S1351324908004944

Roland Hausser's book provides a clear and detailed introduction into the field of database semantics (or DBS). The task of the database semantics project is to provide a model of natural language communication between human agents and robots. There are two main ideas behind DBS. One is the algorithm of left-associative grammar that combines words into sentences by modelling time-linearity. The other is the storage of propositional content in a Word Bank in the form of tables ('proplets').

The book is divided into three parts. The purpose of Part I is to outline the main building blocks of the theory of linguistic communication that governs DBS. Part II examines major constructions of natural language, including functor-argument structure, coordination, and coreference. In Part III, we find concrete suggestions for the software implementation of DBS for the case of the English language. The present review will focus chiefly on Parts I and II.

Chapter 1 details methodological principles of DBS. I shall mention several of them. The 'agent-oriented' approach of DBS treats linguistic signs as results of the speaker's language production. Consequently, DBS is interested in modelling the speaker and the hearer modes of speech. The verification principle states that the theory should face tests by one of its implementations. It is supposed to be different from a similar principle of testability of scientific theories by means of experiments, but it is not entirely clear in what way. The equation principle demands a realistic reconstruction of cognition by means of incremental upscalability of prototypes in terms of their functional completeness and data coverage. Another important methodological assumption is the familiar principle of surface compositionality, according to which the semantic properties of complex expressions should derive systematically from the syntactic category and the literal meaning of the constituent expressions.

The objectivation principle, among other things, postulates the existence of a service channel of communication between the scientist and the artificial agent. The channel allows DBS to avoid the underdetermination of the theory of meaning by linguistic data: the scientist is guaranteed to have privileged access to the 'real' intentions of the artificial cognitive agent (somewhat eccentrically, Hausser calls that problem 'the problem of solipsism'). No eternal observer can in principle have privileged access to the intentions of a human cognitive agent. Clearly this implies a significant distinction between a human agent and an artificial agent. However, a basic assumption of DBS – mentioned in pages 1–2 – is that no such distinction exists.

Chapter 2 presents the SLIM account of communication, the pragmatic theory behind DBS. The speaker meaning of an utterance is determined by its literal meaning and context, where the notion of context is based on the ideas of Montague and Kaplan. The reference of singular terms is initially fixed by a 'private' marker which at a later stage is expected to be correlated with the use of a name in the public language.

Chapter 3 introduces proplets, the primary carriers of propositional content. Proplets are nonrecursive feature structures that contain attributes and their values; however, nested feature structures are not allowed. The reader may easily grasp the idea of a proplet by opening the registry of MS Windows on his personal computer. This chapter also provides useful examples of combining proplets and storing in a Word Bank.

Chapter 4 deals with the DBS approach to meaning. It turns on the distinction between concept types and concept tokens. Concept types are encoded by proplets with constant values, and concept tokens are encoded by proplets that generally may have variable values. Changes in the environment trigger input of parameter values that match the values of a concept type proplet. Then the variable values of the concept type are instantiated by constants. As a result, we obtain a concept token proplet that can be stored in memory. The chapter concludes with more details on action and recognition. One might complain that Hausser's definition of action seems slightly arbitrary. Action is represented in the form of a sequence type-token-periphery. However, it is also possible to see it as a generalised case of recognition, having the sequence periphery-type-token-type-periphery.

In Chapter 5 we find examples of retrieving answers to questions, as well as an interesting discussion of modelling logical inferences within DBS. The way proplets are organised makes them particularly suitable for drawing inferences. However, the choice of sample inferences is eccentric. Modus ponens, in its standard form, is a necessary component of predicate calculus, just as it is of propositional calculus. If one wanted to give examples of typical first-order inferences, one could use universal instantiation or existential generalisation.

Part II begins with Chapter 6 and the representation of intrapositional functor-argument structure. We have examples of representing determiners, adjectives, auxiliaries, prepositions, and passive voice. Again, some care must be exercised in choosing the examples supposedly undermining predicate logic. Thus, the sentence 'Every man loves a woman.' is quite unambiguously formalised in predicate logic, with the universal quantifier taking the widest scope. A better example could be a sentence such as 'There is a special dish for every occasion.'

Chapter 7 covers extrapositional functor-argument structure. The remainder of Part II is devoted to coordination and coreference. Chapters 8–9 deal with intrapositional and extrapositional coordination. There are quite a few useful instances of time-linear derivations for the cases of noun, subject-verb, verb-object, and subject-object coordination. The principles of manipulating proplets are similar in each of the cases, but the wealth of examples allows the reader to have a better grip on particular solutions. Chapter 10 addresses coreference, both intrapositional

and extrapositional. Special attention is paid to the implementation of Langacker-Ross constraint.

Chapters 11–15 of Part III describe programming implementation of the methods of DBS for fragments of English, albeit at a fairly abstract level. Chapters 11 and 12 examine the sequence ‘Julia sleeps. John sings. Susanne dreams.’ in the hearer and speaker modes respectively. The principles of implementation follow straightforwardly from the discussion in Part I. Thus, the task in the hearer mode is essentially in combining isolated proplets ([Julia], [sing], and so forth) of the Word Bank into connected proplets. The fragments examined in chapters 13 and 14 contain complex noun and verb phrases. Finally, in chapter 15 we have fragments containing adnominal and adverbial modifiers.

This is a competent and carefully crafted book. Major claims are patiently laid out. Key notions are explained; however, a certain amount of background knowledge is required. For example, the basic type-token distinction is given via representation of accidental properties. A layman is likely to find such definitions a little baffling. Fortunately, the many examples provided by the author usually dispel the mystery. Typesetting is quite intuitive, and there are surprisingly few typos. The book will be valuable for instructors and graduate students in computer science and linguistics. Advanced undergraduates may also study it with profit.

Dr Sandy Berkovski
Department of Philosophy
Bilkent University
Ankara 06800 Turkey