Ontological Semantics

by

Sergei Nirenburg and Victor Raskin

Preface

This book introduces ontological semantics, a comprehensive approach to the treatment of text meaning by computer. Ontological semantics is an integrated complex of theories, methodologies, descriptions and implementations. In ontological semantics, a **theory** is viewed as a set of statements determining the format of **descriptions** of the phenomena with which the theory deals. A theory is associated with a **methodology** used to obtain the descriptions. **Implementations** are computer systems that use the descriptions to solve specific problems in text processing. Implementations of ontological semantics are combined with other processing systems to produce applications, such as information extraction or machine translation.

The theory of ontological semantics is built as a society of microtheories covering such diverse ground as specific language phenomena, world knowledge organization, processing heuristics and issues relating to knowledge representation and implementation system architecture. The theory briefly sketched above is a top-level microtheory, the ontological semantics theory *per se*. Descriptions in ontological semantics include text meaning representations, lexical entries, ontological concepts and instances as well as procedures for manipulating texts and their meanings. Methodologies in ontological semantics are sets of techniques and instructions for acquiring and manipulating knowledge as well as for running specific applications.

Ontological semantics is not a finished product. It is constantly evolving: new implementations are developed in response to needs for enhanced coverage and utility. Some such needs arise from the internal logic of the field; some others are due to the requirements of practical applications. In any case, ontological semantics is driven by the necessity to make the meaning manipulation tasks, such as text analysis, text generation and reasoning over text meaning representations, work. As a result, our approach places no premium on using a single method, engine or formalism in developing the procedures themselves or acquiring the data that support them. Instead, ontological semantics develops a set of heterogeneous methods suited to a particular task and coordinated at the level of knowledge acquisition and runtime system architecture in implementations.

The methodology of ontological semantics has a hybrid character also because it allows for a variable level of automation in all of its processes—both the runtime procedures and the important knowledge acquisition tasks. Asymptotically, all the processes of ontological semantics will become fully automated. However, to make ontological semantics applicable before it reaches this ideal, human participation must be grudgingly and judiciously built in. In the various extant implementations, the runtime procedures have always been automated, while knowledge acquisition has involved a controlled and channeled human effort. The levels of automation will continue to increase across the board as the approach evolves.

It is arguable that the human level of performance in processing language is a goal that is unattainable by computer programs, either today or, quite possibly, ever. This realization may lead some to rejecting this goal and focusing instead on what is perceived as necessary components of a future NLP system. Often, the focus is further narrowed to methods, formalisms and tools and excludes broad descriptions of phenomena or procedures. Ontological semantics takes the opposite view: it considers the development of implementations and comprehensive applications the main challenge of NLP. We fully realize that, at any given time, these implementations fall short on quality and coverage. While improving specific methods is important, ontological semantics is more interested in developing all the necessary processing and knowledge modules and combining them in a comprehensive system for a class of real-life NLP applications, at the current stage of their attainability.

We appreciate the attraction in setting and reaching attainable local goals, such as the exploration of a single language phenomenon or the perfection of a processing method or a formalism. We are concerned that such efforts are not at all certain to facilitate any future comprehensive systems. When a potential component of a system is developed in isolation, its integrability, if at all considered by the developers, is assumed. Experience in integrating component microtheories in ontological semantics has demonstrated that it is a major resource drain. It follows that minimizing this effort through coordinated development of microtheories is desirable. In practice, of course, there is always a trade-off between importing a microtheory, which would require an integration step, and developing it in house.

Methodological versatility in ontological semantics helps to avoid the fallacy of trying to apply a method of choice to too many tasks. Such misplaced optimism about the utility of any method often results in increased complexity of implementation and/or lower-quality output. In other words, one has to avoid being burned by the old adage, “If all you have is a hammer, everything looks like a nail.” It is a methodological tenet of ontological semantics that every class of phenomena may require a dedicated method. As a result, the approach always addresses options for treatment instead of promulgating the one “correct” way.

Ontological semantics is content-oriented. It puts a premium on acquiring all the knowledge required for the left-hand sides of the many heuristic rules that it uses in processing. Heuristics presuppose abductive reasoning, that is, they are defeasible. The reason for choosing abduction is the realistic expectation that text inputs may at any time violate a recorded constraint in the knowledge base.

The book consists of two parts. In Part I, ontological semantics is positioned vis-a-vis cognitive science and the AI NLP paradigm (Chapter 1), the philosophy of science (Chapter 2), linguistic semantics and the philosophy of language (Chapter 3), computational lexical semantics (Chapter 4) and studies in formal ontology (Chapter 5). Part II describes the content of ontological semantics. Chapter 6 defines and discusses text meaning representation as a process and as a structure. Chapter 7 is devoted to the static knowledge sources in ontological semantics—the ontology, the fact database, the lexicon and the onomasticon. Chapter 8 sketches the ontological semantic processes involved in text analysis. Chapter 9 deals with acquisition of static knowledge in ontological semantics. The content of the various chapters is highly interrelated, which results in a large number of cross-references.

We believe that the book will be of interest to a variety of scholars and practitioners in our field and adjacent areas. NLP specialists and computational linguists will find here a variety of proposals for computational treatment of specific language phenomena; for the content and format of knowledge resources and their acquisition; and for integration of microtheories into a single implementation system. For AI specialists and cognitive scientists, ontological semantics can be seen as a realization and a practical argument for knowledge-based processing in general, for example, within the model of an intelligent agent; it also provides a detailed instance of a complex and multifaceted knowledge base augmented with browsing, acquisition and testing tools. Theoretical linguists (especially semanticists) and the philosophers of language will benefit from exposure to a number of suggested solutions for natural language meaning representation and processing. Descriptive linguists may find specifications for convenient tools to enhance the efficiency of their work. The philosophers of science may find the discussion of the philosophy of linguistics a useful case study for their deliberations. Cognitive psychologists and psycholinguists may wish to consider whether our model of language processing may have any validity for humans; additionally, our system may be considered as a substrate for psychological and psycholinguistic experimentation in human language processing. Specialists in human factors may be interested in the particulars of the division of labor between humans and computers that ontological semantics proposes for knowledge acquisition.

In the area of linguistic engineering and NLP applications, this book may provide a variety of workers with ideas about content and structure of static knowledge sources, about the knowledge requirements of various processors and complete applications, about practical descriptions of particular phenomena, about organizing the knowledge acquisition effort and about integrating comprehensive systems. We also hope that this book will help the practitioners to realize better that: a) treatment of meaning is a *sine qua non* for attaining a new level of quality in practical applications and that the rather steep price for its inclusion is well worth paying; and b) that the crucial component of success of large applications is content, not formalism.

We would like to express our gratitude to our colleagues who, over the years, have contributed to the various implementations and applications of ontological semantics. Allen B. Tucker worked with us on an early conception of knowledge-based machine translation. James H. Reynolds and Irene B. Nirenburg worked on the POPLAR planning application. James Pustejovsky contributed to an early formulation of the microtheory of aspect. Lori Levin collaborated with us on the issue of syntax-driven and ontology-driven semantics. Ira Monarch and Todd Kaufmann were instrumental in building the first acquisition environment for the ontology and an initial formulation of its top levels. Lynn Carlson helped to research the first guidelines of ontological modeling and contributed to the development of an early version of the ontology itself. Salvatore Attardo and Donalee H. Attardo analyzed and catalogued contributions of linguistic semantics to computational applications. Manfred Stede provided programming support for this effort. Ralf Brown helped with a variety of tools and conceptual and procedural support for knowledge acquisition and representation; in particular, he formulated an early version of set notation for ontological semantics. Ingrid Meyer and Boyan Onyshkevych, together with Lynn Carlson, produced an early statement about lexicon structure and content. Christine Defrise contributed to the specification of the format and content of text meaning representation. Ted Gibson implemented the morphological and syntactic analyzers as parts of the Dionysus implementation of ontological semantics. Eric Nyberg implemented the underlying knowledge representation language in which the original ontology was formulated, as well as contributing, with John Leavitt, to the development of the text generator module inside Dionysus. In the Mikrokosmos implementation of ontological semantics, Kavi Mahesh was responsible for the development of the ontology and shared with Steve Beale the work on the semantic analyzer. Steve also worked on the control structure of the implementation as well as on the generation component. Boyan Onyshkevych developed an algorithm for finding the optimum paths between concepts in the ontology, used as the basis of disambiguation in analysis. Evelyne Viegas, Lori Wilson and Svetlana Sheremetyeva provided management and acquirer training support for the knowledge acquisition effort in the Mikrokosmos and CAMBIO/CREST implementations. Eugene Ludovik and Valery Sibirtsev worked on the version of the semantic analyzer for the CAMBIO/CREST implementation. Spencer B. Koehler was responsible for the acquisition tools in this implementation as well as for managing the actual acquisition of the fact database and leading the development of the question answering application in CREST.

We have profited from many discussions, some of them published, of issues in and around ontological semantics with Yorick Wilks, who also read and commented on parts of the manuscript. James Pustejovsky and Graeme Hirst have made useful comments on some of the ideas in the book. Jim Cowie has imparted to us a great deal of his realistic view of NLP in the course of many formal and casual discussions.

There are many people who have, over the decades, have been a source of inspiration and admiration for both or either of us. We have both learned from Igor Melcuk's staunch and fearless refusal to conform to the dominant paradigm as well as his encyclopedic knowledge of linguistics, and ability to mount a large-scale and relentless effort for describing language material. We have always admired Charles Fillmore for having never abandoned an interest in meaning, never sacrificing content for formalism and never refusing to meet the complexities of semantic description head on. We are also grateful to Allen B. Tucker who was a great co-author and enthusiastic supporter in the early days of our joint work in NLP, even before we knew that what we were doing was ontological semantics. We greatly appreciated Jim McCawley's iconoclastic presence in linguistics and mourn his premature death. We agreed early on that Paul M. Postal's treatment of semantic material in his 'remind' article was the early benchmark for our own descriptive work. Roger Schank, a major representative of the 'scruffy' AI tradition of concentrating on the semantic content (no matter how limited the coverage) rather than the formalism, was an important influence. Over the years, we have greatly enjoyed Yorick Wilks' encyclopedic knowledge of philosophy, AI and linguistics as well as his general erudition, so rare in our technocratic times, his energy, his style of polemics, his ever present wit and his friendship. Victor Raskin is forever grateful for the privilege of having worked with Vladimir A. Zvegintsev and Yehoshua Bar Hillel. Sergei Nirenburg would like to thank Victor Lesser for the early encouragement, for the many lessons in how to think about scientific problems, for warmth and wisdom.