HARRY BUNT

FROM LEXICAL ITEM TO DISCOURSE MEANING: COMPUTATIONAL AND REPRESENTATIONAL TOOLS

1. The Span of Computational Semantics

Computing meanings is something that we do when we read, when we write, when we listen and when we speak, and to a certain extent also when we think and when we dream. We seem to do it virtually all the time, and yet it isn't even entirely clear what we *mean* by saying that we compute *meanings;* the very concept of meaning is not beyond discussion.

Computers are superior computing devices; if we could get them to compute meanings in the sense that they associate similar meanings with natural language utterances and texts as people usually do, then that would open fascinating possibilities such as machines that understand what we mean when we talk to them, or machines that can express in ordinary language the regularities that they find in huge amounts of data, or machines that can find exactly those documents and document passages on the internet that are relevant for a particular purpose, based on an understanding of the texts and of the purpose.

In order to realize such possibilities, it would clearly help to have a good understanding of what we mean by 'meaning', and of the kinds of processes and information that are involved in the computations that people carry out when they associate meanings with linguistic objects. Even though we do not know much about the way people compute meanings, there are certain things that we can say on the basis of a conceptual and logical analysis of what it is that we are trying to compute, and of what must necessarily go into the computation.

First, when we compute the meaning of an utterance, two of the main sources of information that we have are the words that make up the utterance and the particular way in which they form the sentence. The intuitively obvious facts that (1) words carry meaning, and (2) the way in which we put words together to form phrases, clauses and sentences (and texts and dialogues) also carries meaning, is the background of the *compositionality* assumption, which says, roughly, that the meaning of a sentence is a function of the meanings of the constituent words and the syntactic structure of the sentence. These two

H. Bunt et al. (eds.), Computing Meaning, Volume 2, 1–10.

sources, lexical meanings and syntactic composition, make up the *linguistic* information that we have at our disposal for computing meanings, together with prosodic information in the case of spoken utterances, and punctuation and layout for textual utterances. The compositionality principle plays a guiding role in formal semantics, and is also useful in computational semantics precisely because it says something (although at a rather abstract level) about the way meanings of complex linguistic objects may be computed from constituent meanings; at the same time it is dangerously restrictive in that it precludes nonlinguistic information to go into the computation of meaning. (See further and Bunt and Muskens, 1999 and Janssen, 1997, for detailed discussions of the principle of compositionality.)

Second, formal semantics has traditionally been preoccupied with studying the meanings of *phrases* and *sentences*, but when we read, speak, listen, or write, we are computing meanings not of words or sentences, i.e. of abstract linguistic constructs, but of concrete objects such as spoken dialogue utterances, movie subtitles, web pages, lines of text on a monitor, and textual elements in an advertisement. The meanings that we compute for such objects are closely tied to the *context* in which they occur, and they are tied to that in a double sense. First, the meaning of, for example, a spoken dialogue utterance has to be something that operates within that context. This is not something like a truth condition or a function from possible worlds to truth conditions, but rather something that relates to the current intentions and beliefs of the dialogue participants. Similarly for web pages and billboard advertisements, where the roles of the designer and the reader may be compared to those of the speaker and hearer in a spoken dialogue. Second, the contextually relevant meaning of an utterance can obviously not be computed solely from the linguistic information in the utterance. There is also linguistic information *outside* the utterance, since a word or a sentence in reality never comes in isolation, but is always part of a larger text or conversation; the meanings that we compute therefore depend strongly on the discourse context. And the information that goes into the computation of utterance meanings is of course not restricted to linguistic information; to understand an utterance in the context in which it occurs one should consider it in relation to other visual and auditory information, such as the graphical elements of a web page, the pictures in an advertisement, the action in the movie that is subtitled, and the gestures and mimics of a speaker in a conversation, and one should also take into account the social and institutional positions of discourse participants and the nature of the discourse that the utterance forms part of. More generally, we always seem to compute context-dependent meanings (or 'interpretations') of linguistic objects by combining linguistic information with nonlinguistic 'world knowledge' of various kinds. In that way we *exploit* the ambiguity and vagueness that natural language expressions have *per se*, and can use natural language as an extremely flexible means of communication. Computing such context-dependent meanings thus means that we have to deal with the entire span from lexical items to context and discourse meanings, which is conceptually enormously wide, and which gives rise to many complex technical issues in computational semantics.

This is a pervasive phenomenon, that emerges virtually everywhere in the computation of meaning, notably in dealing with ambiguity and vagueness. One particular manifestation of this is the phenomenon of *metonymy*, where a speaker says one thing while clearly meaning something else, as illustrated in the following examples:

(1) a. Mary finished the book.

b. Mary enjoyed the book.

The verb 'finish' semantically takes an eventuality as its complement. Depending on whether Mary is known to be a writer, (1a) will be interpreted as *Mary finished writing the book* or as *Mary finished reading the book.* (Other readings are also possible, for instance if Mary is an illustrator, a text editor, a proof reader,..) Similarly, (1b) has as its most obvious interpretation the one where Mary enjoyed reading the book. Such interpretations can be obtained by adding to the lexical item for *book* (or to a supercategory of it) the information that books are meant for reading, and that books are written (and illustrated, edited, proof read, printed,..). If such information is lexically available, then compositional rules can produce the intended readings. However, as Lascarides and Copestake (19987), have argued, if Mary is a goat, then (1a) means that Mary ate the last of the book, and (1b) that Mary enjoyed eating the book - meanings which cannot be obtained from the lexical meanings of the words in the utterances, since the lexicon will not characterize books as being intended to be eaten. This illustrates that real-world knowledge ('goats eat anything, even books') can override purely linguistic knowledge in the computation of meanings. (Lascarides and Copestake 1998 go on to outline a way in which this can be organized, using 'persistent default unification' for the percolation of lexical knowledge and the pragmatic theory DICE for taking world knowledge into account.)

Third, when we say that computing meanings means 'taking world knowledge into account', we are in fact saying that the computation of meaning involves *reasoning*. Reasoning, or 'making inferences', is a process that can take many forms, such as logical deduction, statistically-based pattern recognition, or activity in a neural network. Blackburn *et al.* in their chapter in this book discuss the role of inference in computational semantics, but they forego a definition of 'inference'; it seems clear, though, that the application of grammar rules is typically *not* regarded as a form of inference (although it can be cast in the form of applying an inference system to a set of axioms, as in the 'interpretation-as-abduction' approach of Hobbs et al.; see Hobbs et al., 1983; Hobbs, 2000). The difference between the application of grammatical and lexical rules in a parser/interpreter on the one hand, and the use of general reasoning and world knowledge on the other, seems mainly a matter of the degree in which processes are constrained, general reasoning being less constrained in which pieces of information may be combined to form a correct inference then the process of parsing and interpreting a sentence.

In models of the computation of meaning, intended to apply in meaning computation by computers, *representations* of the input, outputs, and intermediate results in inference processes play a crucial role. This is due to the fact that inference systems based on logical formalisms are tied to particular representations. In computational semantics, representations are important at four levels:

1. at the lexical level, in the representation of the meaning aspects encoded in lexical items;
2. at the grammatical level, to represent the meanings of the combination of words into phrases and clauses;
3. at the discourse level, in representing the contextual meanings of utterances, texts, dialogue contributions, and other concrete linguistic objects;
4. at the level of context, to represent the context information that goes into computing meanings (linguistic, discourse information, as well as nonlinguistic situational information and general world knowledge).

Traditional logical representation formalisms, such as first-order predicate calculus, modal logics, higher-order logics, or typed λ-calculus, can all be said to have *merciless precision*: in contrast to natural language, where ambiguity and vagueness are essential features, the languages of formal logic are traditionally designed to be fully precise and unambiguous. This often creates problems in the application of these formalisms to natural language interpretation, for example because they tend to lead to architectures where disambiguation is forced with a finer 'granularity' then sensible, and at a stage where the information that would be needed for disambiguation is not available, with the effect that spelling out all ambiguity leads to an explosion of possibilities, most of which have to be ruled out at a later stage (see the discussion in Bunt and Muskens, 1999). One of the interesting discoveries of computational semanticists in the last decade is the use of *underspecification* in all kinds of representation (see Bos, 1999; Pinkal, 1999; Bunt and

Muskens, 1999). The use of underspecification allows meaning computation processes to avoid being overly precise and to avoid premature and unnecessary disambiguation. Underspecified representations are currently considered not only at the phrasal and clausal level, but also at the other three levels mentioned above.

2. About this Book

The contributions in this book cluster around four themes, relating to the span of computational semantics as outlined above, and to the computational and representational tools involved in dealing with that span:

1. Lexical semantics
2. Inference
3. Underspecified semantic representation
4. Context and contextual meaning

The first chapter, by **Patrick Blackburn, Johan Bos, Michael Kohlhase and Hans De Nivelle**, discusses inference in computational semantics. The authors argue that state-of-the-art methods in first-order *theorem proving* and *model generation* are of direct relevance to inference for natural language processing. This claim is supported by discussing an implementation by Johan Bos of Rob van der Sandt's presupposition projection algorithm in Discourse Representation Theory, an approach which demands sustained use of powerful inference mechanisms.

The next four chapters are concerned both with inference and with lexical semantics. In their contribution *Building a Semantic Lexicon: Structuring and Generating Concepts*, **Federica Busa, Nicoletta Calzolari, Alessandro Lenci and James Pustejovsky** address the representation of lexical semantic information. One of the main challenges for computational lexical semantics is to bridge the gap between on the one hand theoretical research on the organization of the lexicon and on the formal representation of word meaning, and on the other hand the increasing need of natural language processing systems to access large repositories of lexical knowledge. Starting from some recent extensions of Pustejosvsky's *Generative Lexicon* theory (Pustejovsky 1995; Pustejovsky 1998), the authors present a general model for the development of a set of large-scale lexical resources developed in the context of the SIMPLE project.

They argue that the principles of the Generative Lexicon provide a framework for structuring word meaning which allows for important synergies

PATRICK BLACKBURN, JOHAN BOS, MICHAEL KOHLHASE AND HANS DE NIVELLE

INFERENCE AND COMPUTATIONAL SEMANTICS

1. INTRODUCTION

In this paper we discuss inference in computational semantics. In particular, we argue that state-of-the-art methods in first-order *theorem proving* and *model generation* are of direct relevance to inference for natural language processing. This claim is based on our experience of implementing van der Sandt's approach to presupposition, and much of the paper discusses this application. Incidentally, the reader can experiment with this implementation over the internet: most of what is discussed below is available as part of Johan Bos's DORIS system (Discourse Oriented Representation and Inference System[1]).

This work has its roots in a textbook entitled *Representation and Inference in Natural Language: A First Course in Computational Semantics* (see Blackburn and Bos (2000a) for the latest draft). The goal of this book is straightforward: to present formal semantics from a computational perspective, and equip students with the basic tools required to perform semantic construction computationally. Modularity, reusability, and the use of standard tools is emphasized. Now, as far as *representation* is concerned, it is more or less clear what an introduction to computational semantics should offer: it is obviously sensible to introduce standard semantic representation formalisms such as Discourse Representation Theory (DRT Kamp and Reyle (1993)), to discuss well-known techniques for handling scope ambiguities, and so on. But *inference* is far harder to pin down. What exactly is inference in computational semantics?

Given the present state of knowledge, this is too difficult to answer: "inference" can mean just about anything from issues of architecture design (what information is available for immediate lookup, versus what is to be computed on the fly) to the use of probabilistic techniques. But in spite of this diversity, one topic should arguably play a key role: the use of first-order logic.

Theoretical considerations certainly suggest the importance of first-order inference. Many semantic representation formalisms can be reduced to first-

[1] `http://www.coli.uni-sb.de/~bos/doris`

H. Bunt et al. (eds.), Computing Meaning, Volume 2, 11–28.

order logic (this includes many formalisms which at first glance seem to lie beyond its reach, such as those which make use of partiality, or modal and temporal operators), and even when a full reduction is not possible, first-order logic often provides a useful approximation (a good example is the partial reduction of higher-order logic to first-order logic via generalized models). In particular, as we shall later see, there is a simple reduction from DRT to first-order logic. But first-order inference is not merely of theoretical interest: one of the main points we make in this paper is that it is becoming an increasingly *practical* option.

There is a large and active research community[2] devoted to exploring first-order inference computationally, and a wide range of sophisticated automated theorem provers, model builders, and other tools are now freely available over the internet.

In our view, computational semanticists should take note of these developments; off the shelf tools are now capable of playing a useful role in developing natural language systems with a non-trivial inferential component.[3]

We devote most of this paper to explaining why such tools are relevant to one particular problem: the computational treatment of presupposition. We are going to examine what is arguably one of the most natural (and certainly one of the most empirically successful) approaches to presupposition, namely van der Sandt's DRT based approach (Van der Sandt, 1992). We show how first-order inference techniques can be used to give a simple implementation of van der Sandt's ideas, and suggest that the resulting implementation gives a natural framework for exploring and refining his account. We extract a general lesson from our experiment, and conclude by discussing this.

Restrictions of space force us to assume a certain amount of background knowledge on the part of the reader. In particular, we assume familiarity with the rudiments of DRT (everything the reader needs can be found in the Kamp and Reyle textbook (Kamp and Reyle, 1993), or the first chapters

---

[2] In our experience, this community is interested in natural language applications, and is often prepared to try and accommodate its special needs. In fact, the proof problems generated by the DORIS system discussed below are currently finding their way into the CADE system competition (Sutcliffe and Suttner, 1997) as challenge problems to the automated theorem proving community.

[3] Of course, the idea of using first-order theorem proving techniques for NLP tasks is not new; it's as old as AI itself, and Allen 1995, for example, contains a good textbook level discussion. Nonetheless, few computational semanticists seem aware of developments in contemporary theorem proving and model generation, or of their potential relevance for computational semantics. We think such tools should be a standard part of the computational semanticist's arsenal.

of Blackburn and Bos (2000b)). Furthermore, while we sketch van der Sandt's method, we're going to focus on the inferential aspect of his work, thus it will be useful to have a copy of his classic article to hand; quite apart from its other merits, it's an excellent introduction to many issues in presupposition that we cannot discuss here.

2. Van der Sandt on Presupposition

Van der Sandt gives an *anaphoric* account of presupposition. That is, in his view presuppositions behave much like anaphoric pronouns—in fact the only difference is that presuppositions have more descriptive content. This simple idea has two important consequences. First, there is no need to give an account of presupposition 'cancellation', for there is no such phenomenon; what other accounts regard as a 'cancellation' is simply a case of a presupposition being successfully resolved to an antecedent. Second, because they have descriptive content, presuppositions are sometimes able to 'repair' the context by creating a suitable antecedent; this process is known as *accommodation*.

Van der Sandt expresses his theory in DRT; strictly speaking this is not necessary, but it is certainly advantageous to do so. DRSs are evolving discourse pictures; they display the previously established context, and grow as more information is added. Van der Sandt lets presuppositions contribute a new picture (that is, a new DRS) to this evolving representation, and demands that the new picture be *sensibly* incorporated into the overall representation. Two incorporation mechanisms are permitted. First, presuppositions can be *resolved*, just like ordinary pronouns. The beautiful point about this option is that it calls for no new apparatus: it simply makes use of familiar DRT mechanisms (such as accessibility) for pronoun resolution. Second, presuppositions can be *accommodated*; that is, they can repair the context by creating their own antecedent. Again, this fits beautifully with central ideas of DRT: because presuppositions are associated with DRSs, accommodation is essentially a matter of enlarging part of the picture.

Let's consider two examples, one illustrating resolution, the other accommodation. First some notation. Van der Sandt represents DRSs containing presupposed information by drawing them with dashed lines; we shall use the computationally more convenient convention of prefixing DRSs containing presupposed information with the symbol α (the mnemonic here is that a DRS marked with an α contains *anaphoric* information). We assume that presupposition triggers in the lexicon (such as the definite article, possessive constructions, and proper names) are associated with an appropriate α-DRS.

For our first example, suppose we have already processed the sentence 'A woman snorts'. That is, we have already built the following DRS:

(1)

y
WOMAN(y) SNORT(y)

Suppose the second sentence is 'The woman collapses'. According to van der Sandt, this is what happens. The second sentence, which contains the presupposition trigger 'the', gives rise to the following DRS:

(2)

α: [x | WOMAN(x)] COLLAPSE(x)

(The best way to view this DRS is as an ordinary DRS—but an ordinary DRS marked as being unresolved with respect to presupposed information.) Next we merge this new DRS with the DRS that represents the previous discourse; note that this merging process takes place while the presuppositions are still unresolved. So after merging we obtain:

(3)

y
WOMAN(y) SNORT(y) α: [x | WOMAN(x)] COLLAPSE(x)

Only after merging do we attempt to resolve the presuppositions. We recursively travel through the merged DRS and, for each α-marked DRS we encounter, we try to find a suitable 'anchor' to resolve to. That is, *we try to match the content of the* α*-DRS with that of superordinated DRSs*. Intuitively this is a natural thing to do; after all, presupposed information is supposed to be contextually available.

Let's see how this works. In our example, we only have one elementary presupposition:

(4)

x
WOMAN(x)

Note that if we identify the discourse referents x and y there is a partial match between the outermost DRS and the α-DRS. Carrying out this identification yields:

(5)

y
WOMAN(y) SNORT(y) COLLAPSE(y)

In short, we have successfully dealt with the presupposition induced by 'the', by identifying the discourse referent it introduced with the woman-denoting discourse referent in the preceding context.

That's the basic idea, but things don't always go this smoothly. Sometimes we *can't* find the presupposed information in the preceding context, and resolution is impossible. (Maybe, we missed a bit of a conversation; and anyway, people often have different views about what the assumed context actually is.) To deal with such cases van der Sandt makes use of *accommodation*: if we can't resolve our elementary presuppositions to a suitable element in the context, we don't give up. Instead we simply add the required background information.

Here's an example. Consider the sentence 'If Mia dates Vincent, then her husband is out of town'. Concentrating only on the trigger 'her husband', we get:

(6)

```
+--------------------------------------------------------+
| x y                                                    |
+--------------------------------------------------------+
| x=MIA                                                  |
| y=VINCENT          +--------------------------------+  |
|                    |                                |  |
| +------------+     +--------------------------------+  |
| |            |     |    +-------------------+       |  |
| +------------+ ⇒   |    | z                 |       |  |
| | DATE(x,y)  |     | α: +-------------------+       |  |
| +------------+     |    | HUSBAND(z,x)      |       |  |
|                    |    +-------------------+       |  |
|                    | OUT-OF-TOWN(z)                 |  |
|                    +--------------------------------+  |
+--------------------------------------------------------+
```

Assuming this is the first DRS we have to process (that is, that the DRS built up so far is still empty), there is no candidate DRS for matching the presupposed information that Mia has a husband, which is coded by the following DRS:

(7)

z
HUSBAND(z,x)

FEDERICA BUSA, NICOLETTA CALZOLARI, ALESSANDRO LENCI AND JAMES PUSTEJOVSKY

BUILDING A SEMANTIC LEXICON: STRUCTURING AND GENERATING CONCEPTS

1. INTRODUCTION

One of the main challenges for computational lexical semantics is to bridge the gap between on the one hand theoretical research on the organization of the lexicon and on the formal representation of word meaning, and on the other hand the increasing need by natural language processing (NLP) systems of accessing large repositories of lexical knowledge. The latter, in fact, represents one of the most critical bottle-necks in the development of robust and efficient systems endowed with linguistic intelligence, given the time-consuming and difficult effort required by the construction and maintenance of lexical knowledge bases. This problem has recently been tackled by designing and developing (both at the national and at the international level) general lexical resources based on a common model, with the aim to provide an explicit and (possibly) standard representation of the linguistic content of lexical items at various levels of representation. Actually, their design represents an important challenge for the computational linguistics community, both from the theoretical and the applicative point of view. The difficulty of this enterprise is given by the inherently multi-purpose and domain-independent vocation of such lexical knowledge bases, since they are intended not to be restricted to specific terminological domains or application types, in order to ensure the maximum degree of portability and reusability. In fact, while application needs and domain features set natural constraints on the format of the lexicons, as well as on the type of the information they must contain, similar constraints are not available for the design of a lexicon which aims to provide general linguistic knowledge. In such cases it is crucial to provide satisfactory answers to issues like: What information a computational lexicon must contain? How to represent this information? What is the format to give to a lexical entry?

In this chapter we illustrate the general model for semantic lexicons developed in the EU-sponsored SIMPLE project,[1] which involves the construction

[1] SIMPLE stands for *Semantic Information for Multipurpose Plurilingual Lex-*

H. Bunt et al. (eds.), Computing Meaning, Volume 2, 29–51.

of wide-coverage lexicons for twelve languages,[2] adding a semantic layer to the PAROLE syntactic lexicons. Even though SIMPLE is a lexicon building project, it has also addressed challenging research issues in the realm of lexical semantics grounded on, and connected to, a syntactic foundation. In particular, SIMPLE has tried to answer some of the issues above by largely grounding the architecture of the lexicon on the theoretical approach provided by the Generative Lexicon (henceforth GL), (Pustejovsky, 1995; Pustejovsky, 1998), where *qualia structure* represents the basic building blocks for structuring and generating the concepts expressed by word senses. This way, SIMPLE has benefited from the natural vocation of the GL to deal with the complex architecture of the lexicon, as revealed in both meaning variation and systematic polysemy, thus providing a consistent guidance for the construction, maintenance and customization of lexical entries.

The nature of the questions addressed in this chapter have been shaped by the requirements of SIMPLE: the size of the lexicon to be built, the multilingual framework, the difficulty of establishing a priori the granularity of the semantic representations for all lexical items, the necessity to provide the lexicographers with guidelines for developing consistent representations among all the languages and, finally, the requirement of flexibility and openness towards future extensions geared at particular applications.

The development of such a resource tackles questions that are at the core of lexical semantics research and SIMPLE represents an important opportunity for testing the viability of computational semantics models of lexical knowledge on a large scale and from a multilingual perspective. In particular, this latter aspect requires a careful identification of lexical conceptual structures to be shared by different languages.

More specifically, the following questions are at the core of the SIMPLE model:

A. what is the structure of the concepts expressed by lexical items?
B. how do these concepts differ in terms of their complexity?
C. how is this complexity characterized and by which formal means is represented?
D. do all major categories of the language involve the same conceptual representation?

In the remainder of this chapter, we first discuss some of the standard approaches to computational lexicon design and semantic classification. These

---

icons, sponsored by DG-XIII of EU, within the LE - Language Engineering Programme, and coordinated by Antonio Zampolli.

[2] Catalan, Danish, Dutch, English, Finnish, French, German, Greek, Italian, Portuguese, Spanish, Swedish.

models are then briefly compared to the SIMPLE architecture for lexical entries, which is largely based on the notion of *extended qualia structure*, providing the basic syntax for recursively generating structured conceptual representations.

2. Semantic Types and Lexicon Architecture

Ontologies surely represent a key ingredient in lexical knowledge management, and actually this is the main reason for their renewed fortune in computational lexical semantics. Representing one of the meanings of a word minimally implies (i.) distinguishing it by other senses the same word might have, (ii.) capturing certain inferences which can be performed from it, and (iii.) representing its similarity relations with the meanings of other words. To this purpose, ontologies are powerful formal representational tools exactly because word meanings can actually be regarded as entities to be classified in terms of the ontology types. In this perspective, a given sense can be described by assigning it to a particular type, so that the ontology structure will account for entailments between senses, and similarities between word senses will correspond to the sharing of the same ontology type.

However, as models for the lexicon, ontology design must face an incredibly hard and challenging task, due to the difficulties and complexity of lexical knowledge. This is, in fact, inherently heterogeneous and implicitly structured. Moreover, polysemy is a widespread and pervasive feature affecting the organization of the lexicon. Finally, word senses are multidimensional entities which can hardly be analysed in terms of unique assignments to points in the ontology. As particularly argued in (Pustejovsky, 1995), a suitable type system for lexical representation must be provided with an unprecedented complexity of architectural design, exactly to take into account the 'protean' nature of the lexicon and its multifaceted behaviour, which makes it closer to a kaleidoscope of senses, continually changing their relations and nature depending on the vantage point from which they are observed.

Research in cognitive psychology and lexical semantics has also shown that words crucially differ for the relative salience of different dimensions. For instance, while natural kind terms are mainly organized in terms of taxonomical hierarchies, a proper description of artifactual terms calls for the specification of their function (Keil, 1989). Similarly, different aspects of meaning are to be taken into consideration to provide a suitable representation of the content of abstract terms, verbs, adjectives, etc. Natural language complexity, thus, prevents the adoption of off-the-shelf type systems, and calls for the design of architectures specifically tailored to capture the organization of the lexicon.

In particular, most of the current ontologies are affected by the so-called problem of *ISA–overloading* (Guarino, 1998). The prominent role assigned to the taxonomical ISA relation in the organization of the type system, in fact, lies at the base of important inefficiencies in the representation of word content in crucial areas of the lexicon. The current methodology for building ontologies is mostly centered around the question: *What is a certain entity?* This way, type systems fail to provide efficient representational tools for those word senses which cannot be satisfactorily classified in terms of this semantic dimension. Take for instance the case of words like *priority*, *materials*, *product*, *goal*, *target*, *link*, *mistake*, *dimension*, *member*, etc. The examples below show that the type of entities these nouns express is highly context-dependent, and in many cases very hard to identify too:

(1) a. Factories seemed to be China's highest priority. (*artifact*)

b. Getting food was his main priority. (*eventuality*)

c. The government has changed his priorities. (*abstract_entity*)

(2) a. IBM lauched a new product. (*software*)

b. The product of his best thinking is in the third chapter. (*abstract_entity*)

c. The bombing of Iraq was the product of their diplomatic efforts. (*event*)

(3) John made many mistakes

a. in his paper. (*typos*)

b. in his relationship with Mary. (*eventualities*)

(4) a. In the library I found many materials about the D-Day. (*semiotic_artifacts*)

b. The materials for the roof are in the garden. (*physical_objects*)

c. Please, send me all the *materials* you have about the new project. (*information*)

For instance, something is a *priority* if it is regarded to have precedence over other entities, independently of its specific nature. Similarly, the meaning conveyed by *materials* is exclusively functional, irrespectively of the specific 'substratum' (concrete or abstract) of the entity. Equally functional is the interpretation of *target*: in fact, an entity is a *target* if it fulfills a certain function in a given context. Similarly, anything can be a *link* as long as it connects two entities in a certain way, the specific way, however, being only determined by knowing what those entities are (cf. for instance the semantic difference between the noun phrases *the link between the webpages* and *the link between Rome and Milan*). The essential fact is that trying to characterize the examples in (1)-(4) by multiplying the senses of these nouns according to the different ISA dimensions is not a viable solution, apart from being totally unexplicative. In fact, such nouns can acquire new taxonomical interpretations depending on the linguistic contexts, and are core cases of the generative nature of the lexicon, thus providing important clues of the orthogonal dimensions that intervene in organizing and shaping the conceptual content of lexical items.

Representing such word senses in terms of type systems that rely too much or exclusively on the ISA dimension ends up in lexical characterizations which are often not fully adequate. One interesting example is provided by WordNet, where semantic description is provided by a 'verticalized' taxonomic hierarchy connecting a given synset to a top node. Thus, the backbone of the hierarchy (at least for nouns) is represented by the ISA relation. WordNet, notwithstanding its impressive capacity of structuring the lexicon, fails to offer satisfactory representations for nouns like the ones above, as the characterization of the senses of the noun *part* in Fig. 1 shows.

Notice that a twofold distinction is made: first of all, between *part* as a relation and *part* as an entity, and then between *part* as a concrete, physical object (e.g. a part of a car) and *part* as a psychological feature (e.g. a part of a theory). The problem is that neither of these distinctions is really justified, let alone it justifies the splitting of senses. In fact, a *part* has an inherently relational meaning, and being a part is not a matter of being concrete or abstract, but just of having a certain relation with something else. It is the nature of the entity to which something belongs as a part to determine whether it is abstract or concrete.

Verbs also raise problems for traditional monodimensional analyses. Consider for instance the case of (5). The verbs in (5) show an important analogy with the nouns in (1)-(4). The predicate is so underspecified that under an approach that aims at identifying each sense of a verb in a context with an a priori semantic class – as in (Levin, 1993) – the verb *keep* would not

FRANÇOISE GAYRAL, DANIEL KAYSER AND NATHALIE PERNELLE

IN SEARCH OF THE SEMANTIC VALUE(S) OF AN OCCURRENCE: AN EXAMPLE AND A FRAMEWORK

1. INTRODUCTION

The semantic phenomena discussed in this paper, although pervasive in Natural Language, are often discussed in different frameworks, and therefore their status and importance varies from theory to theory. This paper attempts to sketch a general framework; due to space limitation, only the main ideas will be developed, and we believe that these ideas will be easier to grasp if we concentrate on one word, which will be used in most of our examples. We have selected the french word *examen* (*exam, examination*), but the phenomena under discussion would have been as well described on other usual nouns.

The plan of the paper is as follows: section 2 presents examples of sentences where what could be considered as a single word sense in a lexicon, nonetheless corresponds to a wide set of semantic values; section 3 develops the main features of our framework, namely considering inferences to be central to Semantics, carrying them out at variable depth, taking them to be defeasible; section 4 insists on a specific phenomenon, called 'co-presence' by which, contrary to the usual semantic theories, the occurrence of a word is not interpreted as a single entity; section 5 discusses some consequences for a computational implementation of this framework.

2. EXAMINING *examen*

The word *examen* has roughly at least three main senses: a general observation, e.g. *examen de la situation* (*situation survey*), a medical test, e.g. *vos examens sont bons, vous pouvez rentrer chez vous* (*your tests are fine, you can go back home*), and a school/university exam. All the occurrences of *examen* in this paper concern this last sense.

(1) L'examen aura lieu demain.
The examination will take place tomorrow.

H. Bunt et al. (eds.), Computing Meaning, Volume 2, 53–69.

(2) Paul a laissé l'examen de système sur le bureau de Jean.
Paul left the operating system exam on Jean's desk.

(3) L'examen est trop long. Il ne tient pas sur une page.
The exam is too long. It cannot be written on one page.

(4) L'examen était facile.
The exam was easy.

(5) Cet examen a toujours lieu en juin.
This exam always takes place in June.

(6) L'examen de fin d'année commence mardi.
The final exam begins on Tuesday.

Even if (1)-(6) concern the same sense of the word examen, nevertheless the semantic value of each occurrence is very different. To be brief, the occurrence in (1) refers to the process of testing candidates; in (2), to the paper on which the subject of the exam is written; in (3), to the text of this subject; in (4), to the content of the questions; in (5), to a set of events, one for each year; in (6) to a sequence of events which take place on different days.

These examples (and the list can easily be extended) illustrate the multiplicity of interpretations that a word sense can take according to its context of occurrence. From the point of view of the lexicon, they show that it would be impractical (and probably even impossible) to enumerate all the semantic values that a single word sense might get when the context varies. Symmetrically, from the point of view of the interpretative process, they give evidence for the failure of the classical assumption, that the problem of finding the semantic value of an occurrence amounts to 'disambiguation', i.e. to merely deciding which one, in the enumeration of word senses in the lexicon, is the proper word sense corresponding to that occurrence.

The creative use of words in context - which is pervasive in Natural Language - must be accounted for. This issue is well known, and a lot of recent works (Pustejovsky, 1995; Briscoe and Copestake, 1995; Kayser and Abir, 1995; Ostler and Atkins, 1991; Amghar et al., 1996; Pollard and Sag, 1987; Nunberg, 1995; Pernelle, 1998; Fass, 1991; Hobbs et al., 1988) attempt to compute dynamically the appropriate interpretation rather than assuming a closed list of prespecified senses among which the good choice has to be done.

But that is not exactly our point. As a matter of fact, the above sketchy explanation of all examples (1)-(6) might give the double impression of:

- the existence of a semantic universe, i.e. of an ontology given a priori, which corresponds to a domain sliced into categories (or types) often organized in a lattice in which the semantic interpretation has just to grip the appropriate piece (according to the context). Inferential mechanisms, such as coercion, would then merely improve the basic principle of composition in allowing to reach a category different from the ones which are immediately accessible from the words of the text. Accounting for the previous examples would then require roughly the following categories: 'text', 'information content', 'paper', 'specific-process', 'generic-process','iterative-process' to be accessible from the word *examen*.
- the existence for each sentence of an ultimate semantic representation, representing its meaning, in which each word occurrence has **one** mapping linked to a given category.

In the next section we develop an alternative to these views.

3. SKETCH OF A FRAMEWORK

We think that there cannot exist an universally valid ontology, or set of categories: the interpretation of a text can, within limits, require to create or modify the categories initially known; moreover, the ontology to be used is conditioned by a given task. And even if the task is fixed, it can be performed in many ways, more or less deeply (see 3.2 below). Different parameters, such as the hearer / reader's competences, the situation of enunciation, the communication intention, come into play to determine the 'level' of understanding.

That leads us to consider semantics as essentially based not on **reference** to a prior universe, but on **inference**.

3.1. *Inference: the core of semantics*

The basic fact is that, when humans read a standard text T, they draw more or less the same set of inferences; this can be - and has actually been - experimentally checked: there is a wide agreement among the readers on the conclusions that follow from T. Furthermore, it is noteworthy that the readers make practically no difference between the propositions which are in some sense embodied in T, and those which can only be derived by adding implicit but obvious information. This is so because the readers, while reading T, have access to a shared body of knowledge that we henceforth refer to as **public knowledge**.

So, we consider that interpretation can be considered as the ability to draw inferences from texts. This ability can be judged according to the more specific task consisting in answering some questions about the text.

With this point of view and for illustration, let us reexamine some of the previous examples.

The inferences to be drawn from (1) should include:

(1a) The students who take this examination have probably received some days ago a notification indicating where and when the examination takes place; they expect to receive tomorrow a list of questions, that they are supposed to answer; they will know shortly thereafter whether they have been successful or not.

(1b) An examiner (or an invigilator) will be present tomorrow, ...

From (4):

(4a) Probably, many students have finished before the end of the examination.

(4b) Probably, they will have good marks, ...

Placing the inference at the core of semantics does really matter: other approaches agree on the importance of inferences, but they treat them in a module that operates on a representation which is the result of an inference-less semantic phase, often a translation from natural language utterances to logical forms ; this conception is illusory, since inference is clearly needed at all stages of interpretation; even simple nominal phrases cannot, in spite of their common structure, be given a proper representation without inference, e.g. *the tomorrow examination*, *the math examination*, *the Baccalauréat examination* refer respectively to a time, a subject and a grade.

3.2. *Inferences at variable-depth*

Inferences can be carried on more or less deeply. Consider (6) for example: a rather superficial reading may obtain:

(6a) An interval of time, called final examination, has its beginning point on Tuesday.

(6b) The students who take it will then have finished their year's training programme.

But a lot of other inferences are accessible too, if the need to derive them arises, e.g.:

(6c) The exam period will stretch over several days.

(6d) During this period, the students will not attend lectures.

(6e) During the last days before Tuesday, they are likely to spend most of their time revising their notes, ...

The depth of the required inferences depends on several factors, such as the identity of who utters the sentence, when, to whom. According to the amount of these parameters known to the reader, (s)he will go deeper or not in his/her inferences, particularly to find out the assumed purpose of this utterance. For example, in (2), if *Jean* is known to refer to the invigilator, the question 'Why did Paul leave the exam on Jean's desk?' should yield the answer 'In order for him to make copies of it and hand them out to the students at the beginning of the exam', while otherwise this answer should not be available.

3.3. *Inference rules*

A part of public knowledge is expressed by means of domain-specific inference rules (a) which are expressed using symbols that represent categories (or semantic values). Some other rules (b) relate word occurrences to internal symbols[1] and link these symbols to semantic values.

At the beginning of the interpretation process, only the inference rules (b) that have word occurrences in their premises are applicable; they determine **semantic values** in their conclusion. At that point, inference rules using semantic values in their premises (a) become accessible as well.

Inferences are not drawn once and for all: they can be changed after a deeper analysis and/or after information is found further in the text; as a matter of fact, new information may require to go back (in physically re-reading, or just in rethinking what had been understood);[2] similarly, asking new questions can force the reader to go deeper in his/her interpretation and thereby to realize that some parts were mistaken.

---

[1] Which are of a different kind from the previous ones.

[2] If sentence (4) is followed by 'so we changed the marks scale because we did not want that most candidates have the maximum', we will withdraw inference (4b).

HARRY BUNT AND LEEN KIEVIT

AGENT-DEPENDENT METONYMY IN A CONTEXT-CHANGE MODEL OF COMMUNICATION

1. METONYMY

A well-known problem in the semantic interpretation of natural language is presented by the use of referring expressions to not directly point at their intended referents, but at some associated object. Examples are:

(1) I'm reading Shakespeare.

(2) The London office called.

(3) John works for this newspaper

(4) Turn up the radio.

(5) The buses are on strike.

The phenomenon illustrated by (1) - (5) is known as *metonymy*, which is defined in the Encyclopedia Brittanica (Brittanica, 2000) as follows:

figure of speech in which the name of an object or concept is replaced with a word closely related to or suggested by the original

Metonymic expressions often have a predicate applied to a different type of argument then it expects. The examples (1) - (5) all illustrate this: one doesn't read a person, but something written by that person; an office cannot make phone calls, only someone in the office, and so on.

Hobbs (2001) gives a slightly different definition of metonymy, namely as:

the linguistic device by which an entity is referred to by referring to a functionally related entity.

By requiring a 'functional relation' rather than being 'closely related to or suggested by', Hobbs ties the relation between intended and indicated referents closer, and requires that the relation always delivers a uniquely determined intended referent.

H. Bunt et al. (eds.), Computing Meaning, Volume 2, 71–95.

The idea of a functional relation between intended and indicated referent can be expressed formally as follows. We represent the application of a predicate P to an argument a as $P(a)$, and the 'functional relation' that constitutes a bridge over the type mismatch between predicate and argument as a function f. The modification of an expression E, in which a type mismatch occurs, to form an expression E' in which the type mismatch is resolved, we denote by $E \leadsto E'$. The bridging relation f thus has the effect (6):

(6) $P(a) \leadsto P(f(a))$

The construction of a bridge for dealing with a type mismatch can be expressed in terms of types as follows. Let the predicate P be applicable to objects of type β; P thus has the type of a function from β to the type *truthvalue*: $\mathsf{Type}(P) = (\beta \rightarrow \mathit{truthvalue})$. Let the argument a have type α. The function f then forms a bridge between β and α if it turns objects of type α into objects of type β, i.e. if $\mathsf{Type}(f) = (\alpha \rightarrow \beta)$. In terms of types, (6) corresponds to (7).

(7) $(\beta \rightarrow tv)(\alpha) \leadsto (\beta \rightarrow tv)((\alpha \rightarrow \beta)(\alpha))$

The phenomenon that the interpretation of an expression forces a part of it to be interpreted in a specific way in order to fit the context, is also known as *coercion*, and a function like f in (6) is called a *coercive function*. It may be noted that the representation of coercion in (6) can be read in two ways, since the right-hand side can be read as either (8a) or (8b):

(8) a. $P(\,(f(a)))$

b. $(P \circ f)(a)$

The first of these expresses the view that the argument a is coerced into the argument $f(a)$, to which the predicate P is applied. By contrast, (8b) expresses that the coercive function is used to construct a new predicate $(P \circ f)$, leaving the original argument unaffected. Nunberg (1995) calls the latter form of coercion *predicate transfer*, and suggests that most cases of metonymy should be analysed in that way, rather than as referring to a different argument. He argues that coercion of the argument only occurs when a demonstrative is used, as when a speaker is holding up his car keys and says (9):

(9) This is parked out back

Nunberg provides several arguments supporting the view that (9) is not a case of predicate transfer, such as the strangeness of continuing (9) with (10a), in contrast with the continuation (10b):

(10) a. *It only fits the front door.

b. It may not start at once.

Nunberg (1979) speaks of 'deferred ostentation' in cases like (9); perhaps *'argument transfer'* would be a more appealing terminology, parallelling 'predicate transfer'. We will use the latter terminology.

The claim that most cases of metonymy should be viewed as instances of predicate transfer is supported by the examples (1) - (5) which, different from (9), can all be continued by referring to the *indicated* arguments, as the (a) sentences show, rather than to the *intended* referents, as the (b) sentences witness. (See also Copestake and Briscoe, 1995 for arguments supporting this view.)

(11) I'm reading Shakespeare.

a. Did you see the house where he was born when you were in Stratford?

b. *He is in rather small print, though, which makes the reading less enjoyable.

(12) The London office called.

a. It's the one in Fleet Street.

b. *It didn't say a name.

(13) John works for a newspaper.

a. It's the one that you like to read at breakfast.

b. *It was founded in 1945.

(14) Turn up the radio.

a. It's in the corner behind that plant.

b/ *It's too low for me to hear what they say.

(15) The buses are on strike.

a. They are otherwise quite comfortable.

b. *They want better payment and more security.

We noted, in connection with (6), that the introduction of a coercive function in the semantic representation in itself may not be sensitive to the distinction between predicate transfer and argument transfer; however, in a DRT-style representation of a discourse like the one formed by (9) and (10b), the interpretation of the pronoun *It* in the second sentence may be problematic on a predicate-transfer analysis, where the first sentence would have introduced a discourse referent of the type *key* which would not fit the pronoun in the second sentence.

Although metonymy often involves type mismatches between a predicate and an argument, this is not necessarily the case. The examples (16) and (17) illustrate this.

(16) Aimez-vous Brahms?

(17) The train is getting more expensive next month.

Sentence (16) can be a question about the addressee's amorous feelings toward Mr Brahms or about her appreciation of Brahms' music, depending on the context. The 'product-of' reading may seem more plausible, but if the context is for instance that of a novel about Johannes Brahms and his love affairs, then the 'direct' reading may be more plausible. Similarly, (17) may occur in a conversation where one of the participants is considering to buy a train, and the direct interpretation may be the intended one; in more mundane contexts, where traveling by train is in focus rather than buying a train, the interpretation that train *tickets* are going up is more plausible. In these cases, an interpreter may have to construct a bridge for getting over the type mismatch between a predicate and the *intended*, rather than the *indicated* argument.

The most influential and elaborate computational treatment of metonymy has been provided by Pustejovsky (1993; 1995). This treatment assumes that entries in a lexicon may contain *qualia structures*, a set of entailments associated with the lexical entry. Pustejovsky has proposed four basic *qualia roles*:

1. *constitutive,* relating an object to its parts;
2. *telic,* connecting an object to its function and purpose;
3. *formal,* distinguishing an object within a larger domain;
4. *agentive,* relating an object to objects involved in its creation.

For instance, the lexical entry for *newspaper* might be like in (18), which says that a newspaper is a daily magazine that is created by a company who publishes it:

(18) dailymagazine(x)
agentive: company(z) & publishes(z,x)

Qualia roles can be exploited when during interpretation a type mismatch is detected between the selectional restrictions imposed by a predicate, and one of its arguments.

Suppose for instance that in (3) the predicate *work for* is interpreted as employs(a,b), with the restrictions that a is a company and b is a person. Then dailymagazine(a) is not acceptable as interpretation of *newspaper*, since it is not a company. Following Pustejovsky's approach, we should then use one of the objects associated with the direct referent through some qualia role. In this case we could use company(a), which leads to a logical form like (19):

(19) ∃c newspaper(c) & ∃a company(a) & publishes(a,c) & employs(a,john)

This approach has been shown to be able to handle a wide variety of cases, in particular of instances of what Pustejosvsky has called *logical metonymy,* where additional meaning seems to arise for noun-verb combinations or adjective-noun combinations in a systematic way, as illustrated by (20) and (21):

(20) a. I've finished the book.

b. Philippa enjoyed the book.

(21) a. Richard likes a fast car.

b. Einstein was a passionate violinist.

The general phenomenon that is illustrated by (20) is that a verb that semantically takes an eventuality is combined with an object involved in an eventuality; similarly, (21) illustrates the application of an adjective which semantically applies to eventualities (*fast* to *go*, *passionate* to *play*) to an object involved. In both types of examples it would seem inadequate to postulate different word senses for the verbs and adjectives involved, as Pustejovsky has argued. The use of qualia structures in a single lexical meaning provides a more adequate treatment. There are counterexamples for this approach, however. For instance, as Lascarides and Copestake (1998) have noted, if Philippa in (20b) is a goat, then the intended interpretation of *enjoyed eating the book* cannot be accounted for. (See also Briscoe et al., 1990). Similarly, if Richard in (21a) is an employee in a car demolition company, then the intended interpretation *car that can be demolished quickly* will be unaccounted