Artificial Intelligence 63 (1993) 193–223
Elsevier

ARTINT 1074

Lexical knowledge representation and natural language processing

James Pustejovsky

Computer Science Department, Brandeis University, Waltham, MA 02254, USA

Branimir Boguraev

Speech and Language Technologies, Apple Technologies Inc., Cupertino, CA 95014, USA

Abstract

Pustejovsky, J. and B. Boguraev, Lexical knowledge representation and natural language processing, Artificial Intelligence 63 (1993) 193–223.

Traditionally, semantic information in computational lexicons is limited to notions such as selectional restrictions or domain-specific constraints, encoded in a "static" representation. This information is typically used in natural language processing by a simple knowledge manipulation mechanism limited to the ability to match valences of structurally related words. The most advanced device for imposing structure on lexical information is that of inheritance, both at the object (lexical items) and meta (lexical concepts) levels of lexicon. In this paper we argue that this is an impoverished view of a computational lexicon and that, for all its advantages, simple inheritance lacks the descriptive power necessary for characterizing fine-grained distinctions in the lexical semantics of words. We describe a theory of lexical semantics making use of a knowledge representation framework that offers a richer, more expressive vocabulary for lexical information. In particular, by performing specialized inference over the ways in which aspects of knowledge structures of words in context can be composed, mutually compatible and contextually relevant lexical components of words and phrases are highlighted. We discuss the relevance of this view of the lexicon, as an explanatory device accounting for language creativity, as well as a mechanism underlying the implementation of open-ended natural language processing systems. In particular, we demonstrate how lexical ambiguity resolution—now an integral part of the same procedure that creates the semantic interpretation of a sentence itself—becomes a process not of selecting from a pre-determined set of senses, but of highlighting certain lexical properties brought forth by, and relevant to, the current context.

Correspondence to: J. Pustejovsky, Computer Science Department, Brandeis University, Waltham, MA 02254, USA. E-mail: jamesp@cs.brandeis.edu.

0004-3702/93/$ 06.00 © 1993 — Elsevier Science Publishers B.V. All rights reserved

1. Inadequacies of lexical representations

In this paper we introduce a theory of computational lexical semantics making use of a knowledge representation framework which offers a rich and expressive vocabulary for lexical information. The motivations for this work are at least twofold. Overall, we are concerned with explaining the creative use of language; we consider the lexicon to be the key repository holding much of the information underlying this phenomenon. More specifically, however, it is the notion of a constantly evolving lexicon that we are trying to emulate; this is in contrast to currently prevalent views of static lexicon design, where the set of contexts licensing the use of words is determined in advance, and there is no formal mechanism offered for expanding this set.

The traditional organization of lexicons in natural language processing (NLP) systems assumes that word meaning can be exhaustively defined by an enumerable set of senses per word. Computational lexicons, to date, generally tend to follow this organization. As a result, whenever natural language interpretation tasks face the problem of lexical ambiguity, a particular approach to disambiguation is warranted. The system attempts to select the most appropriate "definition" available under the lexical entry for any given word; the selection process is driven by matching sense characterizations against contextual factors. One disadvantage of such a design follows from the need to specify, ahead of time, the contexts in which a word might appear; failure to do so results in incomplete coverage. Furthermore, dictionaries and lexicons currently are of a distinctly static nature: the division into separate word senses not only precludes permeability; it also fails to account for the creative use of words in novel contexts.

We argue below that the framework for representation of lexical knowledge developed here is superior to current lexical entry formats—both in terms of expressiveness of notation and the kinds of interpretive operations it is capable of supporting. Rather than taking a "snapshot" of language at any moment of time and freezing it into lists of word sense specifications, the model of the lexicon proposed here does not preclude extensibility: it is open-ended in nature and accounts for the novel, creative, uses of words in a variety of contexts by positing procedures for generating semantic expressions for words on the basis of particular contexts.

Adopting such a model presents a number of benefits. From the point of view of a language user, a rich and expressive lexicon can explain aspects of learnability. From the point of view of a natural language processing system, it can offer improvements in robustness of coverage. Such benefits stem from the fact that the model offers a scheme for explicitly encoding lexical knowledge at several levels of generalization. As we discuss below, factoring these along different dimensions makes it possible for NLP systems to carry out semantically intensive—and hitherto complex—tasks. For instance, a consequence of adopting the representation model presented below as the basis of semantic interpretation is that some classically difficult problems in lexical ambiguity are resolved by viewing them from a different perspective. In particular, we illustrate how, by making lexical ambiguity resolution an integral part of a uniform semantic analysis procedure, the problem is rephrased in terms of dynamic interpretation of a word in context; this is in contrast to current frameworks which select among a static, predetermined set of word senses, and do so separately from constructing semantic representations for larger text units.

There are several methodological motivations for importing tools developed for the computational representation and manipulation of knowledge into the study of word meaning, or *lexical semantics*. Generic knowledge representation (KR) mechanisms, such as inheritance structures or rule bases, can—and have been—used for encoding of linguistic information. However, not much attention has been paid to the notion of what exactly constitutes such linguistic information; this has been especially true in the context of developing operational NLP systems. Traditionally, the application area of knowledge representation formalisms has been the domain of general world knowledge. By shifting the focus to a level below that of words (or lexical concepts) we are now able to abstract the notion of lexical meaning away from world knowledge, as well as from other semantic influences such as discourse and pragmatic factors. Such a process of abstraction is an essential prerequisite for the principled creation of lexical entries.

Furthermore, we argue below that judicious use of KR tools enriches the semantics of lexical expressions, while preserving felicitous partitioning of the information space. Keeping lexical meaning separate from other linguistic factors, as well as from general world knowledge is a methodologically sound principle; nonetheless, we maintain that all of these should be referenced by a lexical entry. The mechanisms developed for multi-faceted representation of knowledge in information-rich artificial intelligence contexts facilitate systematic incorporation of world knowledge into the lexicon. On the other hand, these mechanisms also make it possible to maintain the boundary between lexical and common sense knowledge. Additionally, KR tools allow us to concisely state the kinds of generalizations about the systematic patterning of words, which the theory presented below is largely concerned with. In particular, on the basis of a formal language for stating such generalizations, we evolve and propose a set of guidelines for capturing and expressing these within the format of lexical entries.

The interplay of these capabilities—multiple levels of representation for the different kinds of lexical information, systematic reduction of information in the entries, imposing structure both on the lexicon as a whole and on individual entries—offers a novel way of capturing multiple word senses through richer composition. In essence, such capabilities are the base components of a generative language whose domain is that of lexical knowledge. The interpretive aspect of this language embodies a set of principles for richer composition of components of word meaning. In a manner explained later in this paper, semantic expressions for word meaning in context are constructed by a fixed number of generative devices (cf. Pustejovsky [44]). Such devices operate on a core set of senses (with greater internal structure than hitherto assumed); through composition, an extended set of word senses is obtained when individual lexical items are considered jointly with others in larger phrases. The language presented below thus becomes an expressive tool for capturing lexical knowledge, without presupposing finite sense enumeration.

In the remainder of this paper we illustrate a particular theory of computational lexical semantics, which promotes the notion of a *generative lexicon* (Pustejovsky [44]). By way of setting the scene, we discuss certain types of lexical ambiguity, and demonstrate how traditional methods of ambiguity resolution fail to scale up for these (and other) cases. We then develop a model of semantic interpretation embodying richer methods of compositionality, and outline a framework for adequately representing lexical semantics in a knowledge base. We also analyze the effect this has on the size of a lexical knowledge base as a whole, and on the size of individual entries. Issues of organization and content of computational lexicons, and in particular, modeling word meaning via lexical definitions, are directly relevant not only to NLP but also to language acquisition and large-scale lexicon population.

The generative theory of lexical semantics imposes a strong focus on current efforts to derive lexical data from large on-line text resources (dictionaries and corpora): it not only offers a uniform representational framework for expressing the data extracted by the tools and methods of computational lexicography (cf. Boguraev and Briscoe, [10]) but also offers guidance on the kinds of lexical data—or distinctions in the lexical behavior of words—which should be sought in such resources (cf. Boguraev [9], Anick and Pustejovsky [3]). In the final section of this paper we sketch the use of a KR formalism to capture a particular regularity of word behavior—lexical transfer as observed in real data—and make it available to an NLP system.

2. The nature of lexical ambiguity

One of the most pervasive phenomena in natural language is that of ambiguity. This problem confronts language learners and natural language processing systems alike. This is no news: both theoretical and computational linguists are aware of the daunting prospect of accounting for ambiguity. The notion of context enforcing a certain reading of a word, traditionally viewed as selecting for a particular word sense, is central both to global lexical knowledge base design (the issue of breaking a word into word senses) and local composition of individual sense definitions. However, current lexicons reflect a particular "static" approach to dealing with this problem: the numbers of and distinctions between senses within an entry are "frozen" into a fixed system's lexicon. Furthermore, definitions hardly make *any* provisions for the notion that boundaries between word senses may shift with context—not to mention that no lexicon really accounts for any of a range of *lexical transfer* phenomena (cf. Section 5 below).

2.1. Word sense enumeration

There are serious problems with positing a fixed number of "bounded" word senses for lexical items. In a framework which assumes a partitioning of the space of possible uses of a word into word senses, the problem becomes that of selecting, on the basis of various contextual factors (typically subsumed by, but not necessarily limited to, the notion of selectional restrictions), the word sense closest to the use of the word in the given text. As far as a language user is concerned, the question is that of "fuzzy matching" of contexts; as far as a text analysis system is concerned, this reduces to a search within a finite space of possibilities.

This approach fails on several accounts, both in terms of what information is made available in a lexicon for driving the disambiguation process, and how a sense selection procedure makes use of this information. Typically, external contextual factors alone are not sufficient for precise selection of a word sense; additionally, often the lexical entry does not provide enough reliable pointers to critically discriminate between word senses. In the case of automated sense selection, the search process becomes computationally undesirable, particularly when it has to account for longer phrases made up of individually ambiguous words. Finally, and most importantly, the assumption that an exhaustive listing can be assigned to the different uses of a word lacks the explanatory power necessary for making generalizations and/or predictions about how words used in a novel way can be reconciled with their currently existing lexical definitions.

To illustrate this last point, below we present some examples of a problematic nature, both for language learners and for current ambiguity resolution models, as implemented in existing NLP systems.

2.1.1. Creative use of words

Consider the ambiguity and context dependence of adjectives such as *fast* and *slow*, where the meaning of the predicate varies depending on the noun being modified. Sentences (1)–(6) shows the range of meanings associated with the adjective *fast*. Typically, a lexicon requires an enumeration of different senses for such words, to account for this ambiguity:[1]

(1) *The island authorities sent out a fast little government boat, the Culpeper, to welcome us*:
ambiguous between a boat driven quickly and one that is inherently fast.
(2) *a fast typist*:
a person who performs the act of typing quickly.
(3) *Rackets is a fast game*:
the motions involved in the game are rapid and swift.
(4) *a fast book*:
one that can be read in a short time.
(5) *My friend is a fast driver and a constant worry to her cautious husband*:
one who drives quickly.
(6) *you may decide that a man will be able to make the fast, difficult decisions*:
a process which takes a short amount of time.

These examples involve at least four distinct word senses for the word *fast*:

`fast(1)`: moving quickly;
`fast(2)`: performing some act quickly;
`fast(3)`: doing something requiring a short space of time;
`fast(4)`: involving rapid motion.

In an operational lexicon, word senses would be further annotated with selectional restrictions: for instance, `fast(1)` may be predicated by the object belonging to a class of movable entities, and `fast(3)` may relate the action "that takes a little time"—e.g. reading, in the case of (4) above—to the object being modified. Upon closer analysis, each occurrence of *fast* above predicates in a slightly different way. In fact, any finite enumeration of word senses will not account for creative applications of this adjective in the language. For example, consider the two phrases *fast motorway* and *fast garage*. The adjective *fast* in the phrase *a fast motorway* refers to the ability of vehicles on the motorway to sustain high speed, while in *fast garage* it refers to the length of time needed for a repair. As novel uses of *fast*, we

---

[1] Examples here, and in the remainder of the paper, are taken from various corpus sources. These are: *the Birmingham Collection of English Text* (BCET), *Wall Street Journal, 1989* (WSJ), *Readers Digest* (RD), *Longman Dictionary of Contemporary English* (LDOCE).

are clearly looking at new senses which are not covered by the enumeration given above.

Permeability of word senses

Part of our argument for a different organization of the lexicon is based on a claim that the boundaries between the word senses in the analysis of *fast* above are too rigid. Still, even if we assume that enumeration is adequate as a descriptive mechanism, it is not always obvious how to select the correct word sense in any given context: consider the systematic ambiguity of verbs like *bake* (discussed by Atkins *et al.* [6]), which require discrimination with respect to *change-of-state* versus *create* readings, depending on the context (see sentences (7) and (8) respectively).

(7) *John baked the potatoes.*
(8) *Mary baked a cake.*

The problem here is that there is too much overlap in the "core" semantic components of the different readings[2]; hence, it is not possible to guarantee correct word sense selection on the basis of selectional restrictions alone. Another problem with this approach is that it lacks any appropriate or natural level of abstraction. Herskovits [26], in addressing the issue of lexical ambiguity of spatial prepositions, introduces the notion of an *ideal meaning* for a lexical item, which provides the core semantics for the word. These undergo semantic deviations due to convention or pragmatic factors, supplying additional or overriding information to the existing selectional restrictions of the preposition. Thus, from the core meaning of *in*, convention will elicit related but distinct senses for the preposition as used in the two expressions *the hole in the wall* and *the crack in the bowl.*

As these examples clearly demonstrate, partial overlaps of core and peripheral components of different word meanings make the traditional notion of word sense, as implemented in current dictionaries, inadequate (see Atkins [5] for a critique of the flat, linear enumeration-based organization of dictionary entries). Within this approach, the only feasible solution would be to employ a richer set of semantic distinctions for the selection of complements than is conventionally provided by the mechanism of selectional restrictions.[3]

---

[2] Jackendoff [29] correctly points out, however, that deriving *one* core meaning for all homographs of a word form may not be possible, a view not inconsistent with that proposed here.

[3] Others who have addressed the general issue of related word senses and semantic representation are Katz [30], Bierwisch [8], Lakoff and Johnson [34], as well as Talmy [53]. Both Apresjan [4] and Mel'čuk [36] have approached the problem of regularly occurring senses for a word.

2.1.2. *Difference in syntactic forms*

It is equally arbitrary to create separate word senses for a lexical item just because it can participate in several subcategorization forms; yet this has been the only approach open to computational lexicons that are based on a fixed number of features and senses. A striking example of this is provided by verbs such as *believe* and *forget.* The sentences in (9)–(13) show that the syntactic realization of the verb's object complement determines how the phrase is interpreted semantically. The *that-complement*, for example, in (9) exhibits a property called "factivity" (Kiparsky and Kiparsky [31]), where the object proposition is assumed to be a fact regardless of what modality the whole sentence carries. Sentence (12) contains a "concealed question" complement (Grimshaw [23]), so called because the phrase can be paraphrased as a question. These different interpretations are usually encoded as separate senses of the verb, with distinct lexical entries.

(9) *Madison Avenue is apt to forget that most folks aren't members of the leisure class.*
A factive reading.

(10) *But like many others who have made the same choice, he forgot to factor one thing into his plans: Caliphobia.*
A non-factive reading.

(11) *As for California being a state being run by liberal environmental loonies, let's not forget where Ronald Reagan came from.*
An embedded question.

(12) *What about friends who forget the password or never got it?*
A concealed question.

(13) *He leaves, forgets his umbrella, comes back to get it ....*
Ellipsed non-factive.

Sensitivity to factivity would affect, for instance, the interpretation by a question-answering system: when asked *Did Mary lock the door?*, depending on whether the input was *Mary forgot that she locked the door* (factive), or *Mary forgot to lock the door* (non-factive), the answers should be *Yes* and *No* respectively. Such a distinction could be easily accounted for by simply positing separate word senses for each syntactic type, but this misses the obvious relatedness between the two instances of *forget.* It also misses not only the parallel between the question-like readings in (11) and (12), but also the similarity between the non-factive in (10) and the ellipsed non-factive in (13). Moreover, the general "core" sense of the verb *forget*, which deontically relates a mental attitude with a proposition or event, is lost between the separate senses of the verb. A more elegant theory would have one definition for *forget* which could, by suitable composition with the different complement types, generate all the allowable readings (cf. Pustejovsky [46]).

2.2. Towards a dynamic model of the lexicon

The major thrust of the preceding analysis has been to outline how the ambiguities shown above cannot be adequately handled by exhaustive enumeration of what are regarded as different word senses. It follows that the conventional computational framework for lexical ambiguity resolution, and in particular, the format for lexical entries in current computational lexicons, fails in at least two respects. First, it is unable to describe all the "senses" of a word through finite enumeration; second, it is also unable to capture interesting generalizations between senses of the same word.

Such failures are partially caused by limited lexical knowledge made available to natural language processing systems, as well as to an impoverished notion of lexical inference. Thus, the traditional framework for ambiguity resolution only employs "static" knowledge, expressed as selectional restrictions, and no specific knowledge manipulation mechanisms apart from the simple ability to match valences of connected words. In contrast, we show below how a lexical entry can be assigned a richer knowledge structure and how, by performing specialized inference over the ways in which aspects of knowledge structures of words in context can be composed, mutually compatible and relevant lexical components of words and phrases are highlighted. This process, licensed by constraints operating through the inference mechanisms, in fact, results in generating a semantic interpretation of a phrase, resolving *en route* the ambiguity of lexical items at their source.

3. Ambiguity and compositionality

The richer structure for the lexical entry proposed here takes to an extreme the established notions of *predicate–argument structure, primitive decomposition* and *conceptual organization*; these can be seen as determining the space of possible interpretations that a word may have. That is, rather than committing to an enumeration of a pre-determined number of different word senses, a lexical entry for a word now encodes a range of representative aspects of lexical meaning (cf. Section 3.1 below). As we will demonstrate, for an isolated word, these meaning components simply define the semantic boundaries appropriate to its use. When embedded in the context of other words, however, mutually compatible roles in the lexical decompositions of each word become more prominent, thus forcing a specific interpretation of individual words within a specific phrase. It is important to realize that this is a generative process, which goes well beyond the simple matching of features. In fact, this approach requires, in addition to a flexible notation for expressing semantic generalizations at the lexical level, a mechanism for composing these individual entries on the phrasal level.

The emphasis of our analysis of the distinctions in lexical meaning is on studying and defining the role that *all* lexical types play in contributing to the overall meaning of a phrase. This is not just a methodological point: crucial to the processes of semantic interpretation which the lexicon is targeted for is the notion of *compositionality*, necessarily different from the more conventional pairing of verbs as functions and nouns as arguments. As we indicated earlier, if the semantic load in the lexicon is entirely spread among the verb entries, as many existing computational systems assume, differences like those exemplified in (7)–(8) and (9)–(13) can only be accounted for by treating *bake*, *forget*, and so forth as polysemous verbs. If, on the other hand, elaborate lexical meanings of verbs and adjectives could be made sensitive to components of equally elaborate decompositions of nouns, the notion of spreading the semantic load evenly across the lexicon becomes the key organizing principle in expressing the knowledge necessary for disambiguation.

To be able to express the lexical distinctions required for analyzing the examples in the last section, it is necessary to go beyond viewing lexical decomposition as based only on a pre-determined set of primitives; rather, what is needed is to be able to specify, by means of sets of predicates, different levels or perspectives of lexical representation, and to be able to compose these predicates via a fixed number of generative devices. The "static" definition of a word provides its literal meaning; it is only through the suitable composition of appropriately highlighted projections of words that we generate new meanings in context. This position is amply illustrated below.

Our theory overcomes the shortcomings, in particular those from the perspective of automatic language processing, of the descriptive, exhaustive enumeration of word senses. What makes this possible is the combination of two notions, both of them following from general principles of KR. First, by incorporating in this language a set of rules governing the generative processes which apply to different levels of word meanings, we are no longer confined to the constraints which follow from operating with a fixed inventory of primitives; namely, not being able to account for the completeness or creative aspects of language (cf. Wilks [57]). Secondly, through the very nature of these rules, we are assured that the semantic representations ultimately associated with text fragments are going to be well-formed. This model proves to be more useful for NLP than existing theories, because lexical disambiguation is treated no differently from the process of semantic interpretation itself.

3.1. Levels of lexical meaning

Following an analysis of a broad range of ambiguous constructions, and in particular of those aspects of word meanings which account for the ambiguities, Pustejovsky [44] argues that in order to address a range of common lexical phenomena, a theory of computational lexical semantics needs to make reference to four levels of representations:

- *Argument structure* encodes the conventional mapping from a word to a function, and relates the syntactic realization of a word to the number and type of arguments that are identified at the level of syntax and made use of at the level of semantics (cf. Grimshaw [24]).
- *Event structure* identifies the particular event type for a verb or a phrase. There are essentially three components to this structure: the primitive event type—state (**S**), process (**P**) or transition (**T**); the focus of the event; and the rules for event composition (cf. Allen [2], Moens and Steedman [37], Passonneau [42], Pustejovsky [45]).
- *Qualia structure* defines the essential attributes of objects, events, and relations, associated with a lexical item. By positing separate components (see below) in what is, in essence, an argument structure for nominals, nouns are elevated from the status of being passive arguments to active functions (cf. Moravcsik [40], Wilks [56], Schank [51], Fillmore [21]). We can view the fillers in qualia structure as prototypical predicates and relations associated with this word (cf. also Mel'čuk [36]).
- *Lexical inheritance structure* determines the ways in which a word is related to other words in the lexicon. In addition to providing information about the organization of a lexical knowledge base, this level of word meaning provides an explicit link to general world (commonsense) knowledge (cf. Quillian [50], Woods [59], Touretzky [54]).

A set of generative devices connects the four levels, providing for the compositional interpretation of words in context. The most important of these devices is a semantic transformation called *type coercion*—analogous to coercion in programming languages—which captures the semantic relatedness between syntactically distinct expressions. As an operation on types within a λ-calculus, type coercion can be seen as transforming a monomorphic language into one with polymorphic types (cf. Cardelli and Wegner [16], Klein and van Benthem [32]). Argument, event, and qualia types must conform to the well-formedness conditions defined by the type system and the lexical inheritance structure when undergoing operations of semantic composition. Lexical items are strongly typed yet are provided with mechanisms for fitting to novel typed environments by means of type coercion over a richer notion of types (cf. Section 3.3 below).

Since the only level of lexical representation not extensively discussed in the literature is that of qualia structure, we briefly outline its components below.

3.2. Qualia structure

Qualia structure is a system of relations that characterizes the semantics of nominals, very much like the argument structure of a verb (Pustejovsky [44]).

In effect, the qualia structure of a noun determines its meaning in much the same way as the typing of arguments to a verb determines its meaning. The elements that make up a qualia structure include familiar notions such as container, space, surface, figure, or artifact. These components of an object's denotation have long been considered crucial for our commonsense understanding of how things interact in the world (cf. Hayes [25], Hobbs et al. [28], and Croft [19]).

Briefly, the qualia structure of a word specifies four aspects of its meaning:

- the relation between an object and its constituent parts;
- that which distinguishes it within a larger domain;
- its purpose and function;
- factors involved in its origin or "bringing it about".

These aspects of a word's meaning are called its *Constitutive role*, *Formal role*, *Telic role*, and *Agentive role*, respectively. Some of these roles are reminiscent of descriptors used by various researchers, such as Wilks [56], Hayes [25], and Hobbs et al. [28]. Within the theory outlined here, these roles determine a minimal semantic description of a word which has both semantic and grammatical consequences. The motivation for positing such characterizations of word meaning is that by enriching the semantic descriptions of nominal types, we are able to "spread the semantic load" more evenly through the lexicon, while accounting for novel word senses arising in syntactic composition. These factors help to structure the lexical knowledge in the lexicon from different perspectives.

It should be pointed out that the construction of concept lattices and ontological hierarchies in artificial intelligence (AI) has proved beneficial for several reasons (cf. Brachman and Schmolze [12] and Carbonell [15]). Chief among these is perhaps their role in simplifying rules of inference as well as facilitating learning. In computational linguistics, however, extensive encoding of information associated with words beyond their logical behavior has been largely avoided and seen as delving into the murky area of commonsense knowledge. Although these fears are ill-founded, some of the methodological points from linguistic analysis are helpful for evaluating competing representational formalisms. For example, by trying to link representations to observable patterns of linguistic behavior in the language, we avoid *ad hoc* formulations and constructs (cf. Moens et al. [38]).

There are distinct linguistic motivations for representing objects in these terms. For example, there are well-known cases of *container–containee* and *figure-ground* ambiguities, where a single word may refer to two aspects of an object's meaning (cf. Apresjan [4], Wilks [56], Herskovits [26], Lakoff [33], and Pustejovsky and Anick [48]). The words *window, door, fireplace,* and *room* can be used to refer to the physical object itself or the space associated with it:[4]

(14) (a) *They walked through the door.*
     (b) *She will paint the door red.*
(15) (a) *Black smoke filled the fireplace.*
     (b) *The fireplace is covered with soot.*

The semantics of the objects mentioned above can be interpreted as a relation between the *Formal* and *Constitutive* qualia. For example, the qualia structure for the noun *door* makes explicit reference to both these aspects of its meaning, and specifies the purpose or use of the object in the value for the *Telic* role.

$$\begin{bmatrix} \textbf{door(x,y)} \\ \text{CONST} = \textbf{aperture(y)} \\ \text{FORMAL} = \textbf{physobj(x)} \\ \text{TELIC} = \textbf{walk\_through(P,w,y)} \\ \text{AGENTIVE} = \textbf{artifact(x)} \end{bmatrix}$$

The noun *door* is represented as a relational type with two parameters, where each parameter is defined in the qualia roles. The above structure makes it explicit that the concept associated with a door is a relation between an aperture of some sort, y, and the physical object itself, x. The *Telic* role refers to an event type of **Process**, and an individual **w** walking through the aperture of the door. Because we have specified explicitly what the subtypes of the noun are, the rules of semantic composition and selection can make use of them. Thus, as we shall see in the next section, the appropriate sense of the noun is selected by reference to qualia information. Many other cases of lexical ambiguity can be analyzed as logical relations within a richer representational framework such as this (see Section 3.3).

Another way of modeling the qualia structure computationally is as a set of constraints on types. For example, Copestake and Briscoe [18] model the general ideas of Generative Lexicon Theory in terms of a type system for a lexical knowledge base (cf. Copestake [17] for details). The operations in

---

[4] It should be pointed out that the classic ambiguities involving the two senses of *bank* and *buck* are not at issue here. These are cases of what Weinreich [55] called contrastive ambiguity, and do not involve aspects of the same concept. See Hirst [27] for discussion of these issues, however.

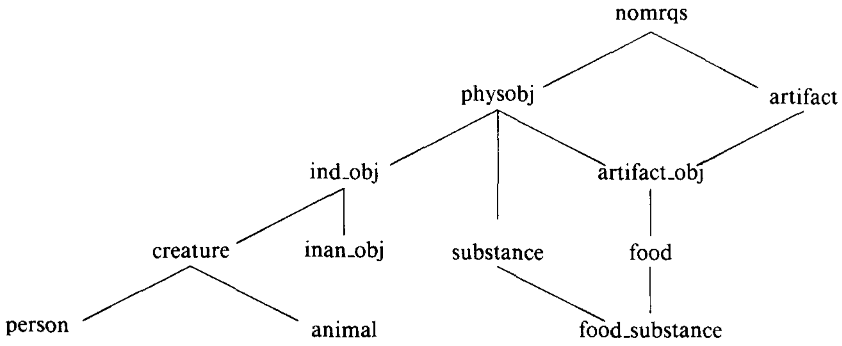

Fig. 1. Fragment of a type hierarchy (from Copestake and Briscoe [18]).

the compositional semantics make reference to the types within this system. The qualia structure along with the other representational devices (event structure and argument structure) can be seen as providing the building blocks for possible object types. Figure 1 illustrates a type hierarchy fragment for knowledge about objects, encoding qualia structure information.[5]

Constraints are seen as restrictions on PATR-II-type feature structures (cf. Shieber [52] or Pollard and Sag [43]). Thus, the constraints for the types **artifact** and **physobj** are given below, where only partial listings for the qualia have been given for purposes of illustrating inheritance.

$$\begin{bmatrix} \textbf{artifact(x)} \\ \text{TELIC} \ = \ \textbf{Pred(E,y,x)} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{physobj(x)} \\ \text{FORMAL} \ = \ \textbf{physform(x)} \\ \text{PHYSICAL-STATE} \ = \ \textbf{solid(x)} \end{bmatrix}$$

Following Copestake and Briscoe [18], bold face indicates types. For example, **Pred(E,y,x)** is of type **formula**, and denotes the set of relations (with event type E) which refer to the use or purpose of x. Any feature structure of type **artifact** must have a feature structure of type **formula** as the value for its TELIC feature. Notice that the constraint on **artifact_obj** contains information inherited from both parents:

---

[5] In Fig. 1, the term, nomrqs, refers to a "relativized qualia structure", a type of generic information structure for entities (cf. Calzolari [14] for discussion). Further, ind_obj represents "individuated object".

$$\begin{bmatrix} \textbf{artifact\_obj(x)} \\ \text{FORMAL} = \textbf{physform(x)} \\ \text{PHYSICAL-STATE} = \textbf{solid(x)} \\ \text{TELIC} = \textbf{Pred(E,y,x)} \end{bmatrix}$$

From type fragments such as these, it becomes clear how a sentence such as *Mary finished her sandwich* receives the default interpretation it does: the noun *sandwich* contains information of the "eating activity" as a constraint on its *Telic* value, due to its position in the type structure; that is, **eat(P,w,x)** denotes a process, **P**, between an individual **w** and the physical object **x**.

$$\begin{bmatrix} \textbf{sandwich(x)} \\ \text{CONST} = \{\textbf{bread,...}\} \\ \text{FORMAL} = \textbf{physobj(x)} \\ \text{TELIC} = \textbf{eat(P,w,x)} \\ \text{AGENTIVE} = \textbf{artifact(x)} \end{bmatrix}$$

We return to examples such as this below, and explain how they are interpreted compositionally.

Having outlined the basic structure of a framework for representing relational information about objects, we will show how qualia structure enriches the semantic description of nominals so that the rules of semantic composition may make direct reference to this information.

3.3. Lexical ambiguity and compositionality

In this section, we examine how this model of lexical structure is able to account for the ambiguities discussed earlier. Consider first the examples with the adjective *fast* (cf. Section 2.1). We can capture the general behavior of how such adjectives predicate by making reference to the richer internal structure for nominals suggested above. That is, we might view *fast* as always predicating of the *Telic* role of a nominal, since the *Telic* role is always typed as an event. To illustrate this, consider the qualia structure for a noun such as *car*:

$$\begin{bmatrix} \textbf{car(x)} \\ \text{CONST} = \{\textbf{body,engine,...}\} \\ \text{FORMAL} = \textbf{physobj(x)} \\ \text{TELIC} = \textbf{drive(P,y,x)} \\ \text{AGENTIVE} = \textbf{artifact(x)} \end{bmatrix}$$

Notice that the *Telic* role specifies the purpose and function of the noun. In the phrase, *a fast car*, it is the relation `drive`, seen as an event, namely, a process, P, which is modified by the adjective as being fast. Similarly, for the nouns *typist*, *waltz*, *book*, and *reader*; it is their *Telic* role that is interpreted as being fast. Without going into details, we note here that the *Telic* role of *typist* or *reader* determines the activity being performed, namely typing or reading; similarly for *waltz*, its *Telic* role refers to dancing. In the case of *book*, the *Telic* specifies common activities like reading or writing. Hence, the interpretations of *fast* in the examples (1)–(6) above can all be derived from a single word sense, and there is no need for enumerating the different senses (cf. Pustejovsky [47]). The lexical semantics for this adjective will indicate that it acts as an event predicate, modifying the activity which is the value of the *Telic* role of the noun. Notice that, in addition to obviating the need for separate senses, we can generate the novel use of *fast* mentioned above in the phrase *a fast motorway*, since the *Telic* role of *motorway* specifies its purpose, and it is this process, P, which is interpreted as fast:

$$[\mathit{Telic}: \mathit{travel}(P, \mathit{cars}) \wedge \mathit{on}(P, x)].$$

The composition of the expression defining *fast* with the lexical aspect it specifies as its "target"—the *Telic* role of its argument (*motorway*)—results in an interpretation corresponding to a use of the word when referring to a road: one that allows for fast travel by cars:

$$\lambda x[\mathit{motorway}(x) \ldots [\mathit{Telic}: \mathit{travel}(P, \mathit{cars}) \\ \wedge\ \mathit{on}(P, x) \wedge \mathit{fast}(P)] \ldots].$$

The notion that a word can specify a target type for its argument is a very useful one, and intuitively explains the different syntactic argument forms for the verbs below. In sentences (16) and (17), noun phrases and verb phrases appear in the same argument position, somehow satisfying the type required by the verbs *enjoy* and *begin*. In sentences (18) and (19), noun phrases of very different semantic classes appear as subject of the verbs *kill* and *wake*.

(16) (a) *Mary enjoyed the movie.*
(b) *Mary enjoyed watching the movie.*
(17) (a) *Mary began a book.*
(b) *Mary began reading a book.*
(c) *Mary began to read a book.*
(18) (a) *John killed Mary.*
(b) *The gun killed Mary.*
(c) *The bullet killed Mary.*
(19) (a) *The cup of coffee woke John up.*
(b) *Mary woke John up.*
(c) *John's drinking the cup of coffee woke him up.*

If we analyze the different syntactic occurrences of the above verbs as separate lexical entries, following the sense enumeration theory outlined in previous sections, we are unable to capture the underlying relatedness between these entries; namely, that no matter what the syntactic form of their arguments, the verbs seem to be interpreting all the phrases as events of some sort. It is exactly this type of complement selection which we will refer to as *type coercion* (Pustejovsky [44]).

How could such a semantic mechanism as that mentioned before for *fast* and the above verbs be represented computationally? As mentioned above, the tools needed to capture the generative nature of word senses are generally available within a typed logic with standard function application and composition (as presented, for instance, by Montague [39] and Aït-Kaci [1]), with some important modifications and extensions.

Perhaps the most important component of Generative Lexicon Theory is the set of devices which give rise to the generative production of new word and phrase senses. These rules of coercion presuppose a typed ontology such as that outlined in the previous section.[6] By allowing lexical items to coerce their arguments, we obviate the enumeration of multiple entries for different senses of a word. We define coercion as follows (cf. Cardelli and Wegner [16]).

Type coercion. A semantic operation that converts an argument to the type which is expected by a function, where it would otherwise result in a type error.

We assume a standard definition for function application:

Function application. If α is of type $< b, a >$, and β is of type b, then $\alpha(\beta)$ is of type a.

Assume that each expression α has available to it, a set of shifting operators, Σ_α, which operate over the expression, changing its type and denotation. We will refer to this set as the expression's *aliases*. By making reference to an expression's aliases directly in the rule of function application, we can treat the item polymorphically, as below:

Function application with coercion (FA$_C$). If α is of type $< b, a >$, and β is of type c, then

(i) if type $c = b$, then $\alpha(\beta)$ is of type a;

---

[6] Types are identified as the intensional types within a typed λ-calculus, and sorts are defined within types for purposes of semantic refinement or specialization as given in the above lattice (cf. Beierle et al. [7]).

(ii) if there is a $\sigma \in \Sigma_\beta$ such that $\sigma(\beta)$ results in an expression of type b, then $\alpha(\sigma(\beta))$ is of type a;
(iii) otherwise a type error is produced.

In order to be able to refer to *segments* of a type-lattice, and not just a node, we will use the notion of a *type-path*, defined below, where $\leqslant$ has the standard lattice-theoretic interpretation:

(i) If a is a type, then $[a]$ is a type-path.
(ii) If a and b are type-paths, and $b \leqslant a$ then $[a\ b]$ is a type-path.

Then, if we extend the set of types to include explicit reference to type-paths, we arrive at the following definitions for types (where e and t are the standard types from Montague [39]):

(i) $[e]$ is a type.
(ii) $[t]$ is a type.
(iii) If $[a]$ and $[b]$ are any types, then $<[a],[b]>$ is a type.

We can then define further coercion operations which affect specific feature structures in the type structure. For example, subtype coercion can be defined as follows:

Subtype coercion. A semantic operation that converts an argument to the subtype which is expected by a function, where it would otherwise result in a subtype error (e.g. selectional violation).

Now assume for each expression α, that there are transformational operators, Σ'_α, which operate over an expression, changing its subtype within a type. We state the polymorphic rule for application below:

Function application with subtype coercion (FA$_{\mathbf{SC}}$). If α is of type $<[b\ c],[a]>$, and β is of type $[b\ d]$, then
(i) if type $c = d$, then $\alpha(\beta)$ is of type a;
(ii) if there is a $\sigma \in \Sigma'_\beta$ such that $\sigma(\beta)$ results in an expression of sort c, then $\alpha(\sigma(\beta))$ is of type a;
(iii) otherwise a subtype error is produced.

Returning to the examples (16)–(19) above with multiple complement selection, we can now offer a solution for why a verb like *begin* is able to select an NP such as *a book* in example (17) above. The qualia structure for *book* contains at least the information that there is a *Telic* event associated with it, whose value is *reading*.

$$\begin{bmatrix} \textbf{book(x)} & \\ \text{FORMAL} = \textbf{physobj(x)} & \\ \text{TELIC} = \textbf{read(P,y,x)} & \\ \text{AGENTIVE} = \textbf{write(T,w,x)} & \end{bmatrix}$$

In the verb phrase *begin the book*, the verb *begin* expects a phrase whose semantic type is an event. Because the NP *the book* does not satisfy this type, the verb coerces the NP into an event denotation, one which is available from the head's own qualia structure. Thus, formally, each qualia aspect is a partial function from noun denotations into one of their subconstituents (cf. Pustejovsky [46] for details). The verb *begin*, therefore, can be said to semantically select for an argument of type `[event]`, instead of requiring three syntactic subcategorization frames. The information that an NP of type `[individual physobj]` can be coerced into an event comes from the set of aliases associated with the noun *book*. Notice that there are two event types associated with **book** through the qualia roles. Aliases for a lexical item are inherited from particular qualia structures for that object. For example, any lexical structure with a *Telic* constraint specified will inherit the type of that constraint as an alias. This, then, gives us a truly polymorphic treatment of verbs such as *begin*, due to coercion and qualia structure.[7]

We finally return to the behavior of adjectival modification discussed above, such as *the fast car*. The feature-based lexical definition for **car** given at the beginning of this section can more formally be expressed as:

$$\begin{aligned} \lambda x[car(x) \wedge\ & Const(x) = \{body, engine, \ldots\} \\ \wedge\ & Formal(x) = physobj(x) \\ \wedge\ & Telic(x) = \lambda y, e[drive'(x)(y)(e)] \\ \wedge\ & Agentive(x) = \lambda y, e[create'(x)(y)(e)]]. \end{aligned}$$

For our present purposes, we abbreviate these functions as Q_F, Q_C, Q_T, and Q_A. When applied, they return the value of a particular qualia role. For example, the purpose of a car is for driving, it comes about by being created, and so on.

$$\begin{aligned} Q_T(car) &= \lambda x \lambda y[drive(x)(y)] \\ Q_A(car) &= \lambda x \lambda y[create(x)(y)]. \end{aligned}$$

Certain modifiers can be seen as modifying only one or a subset of the qualia for a noun, resulting in a type of restricted modification. Formally, we can accomplish this by making reference to the system of subtyping mentioned

---

[7] What actually drives the type coercion is not entirely clear. For example, since *begin a book* seems to be able to refer to either the "reading" or the "writing" events, it may, in fact, be merely event type satisfaction and not qualia satisfaction.

above; that is, an element is actually a set of type features structured as a semilattice.

This allows us to go beyond treating adjectives such as *fast* as intersective modifiers, for example, $\lambda x\,[car'(x) \wedge fast'(x)]$. Let us assume that an adjective such as *fast* is a member of the general type $<[N],[N]>$, but can be subtyped as applying to the *Telic* role of the noun being modified, as argued above. That is, it has as its type, $<[N, Telic],[N]>$. The interpretation of the partial noun phrase, *fast car* can now be given below:[8]

$$\begin{aligned}\lambda x\,[car(x) \;&\wedge\; Const(x) = \{body, engine, \ldots\}\\ &\wedge\; Formal(x) = physobj(x)\\ &\wedge\; Telic(x) = \lambda y, e\,[drive'(x)(y)(e) \wedge fast(e)]\\ &\wedge\; Agentive(x) = \lambda y, e\,[create'(x)(y)(e)]].\end{aligned}$$

Given that types can refer to type features, we can treat restrictive modification as function application with coercion in the following manner. Following our previous discussion, we assume that a function can coerce its argument into a specified type just in case there is an alias σ which, when applied to the argument, gives the desired type. Notice from the example discussed above, *fast car*, that the *Telic* interpretation of *fast* is only available because the head has a *Telic* value specified. This indicates that for the noun type N, a type feature $[N, Telic]$ is available as an inclusion polymorphism. Therefore, we can treat the semantics of such restrictive modification as follows: if α is of type $<[N,Q],N>$, and β is of type N, then $[\![\alpha\beta]\!] = \beta \cap \alpha(Q_\beta)$. Thus, the role played by a type feature is to allow a composition to be well-formed, while restricting the scope of the denotation of the adjective.

In this section, we have shown how to resolve lexical ambiguity through the rules of semantic composition directly, rather than keeping lexical selection a separate process, divorced from the interpretive process.

4. Knowledge representation and lexical organization

So far we have looked at the "classical" problem of ambiguity of words, manifested in the problem of how to select suitable word senses for a word in running text, according to some notion of context. The solution outlined in the previous section, in addition to offering an alternative way of approaching the problem, has the important effect of reducing the size of the lexicon.

---

[8] Bierwisch [8] proposes a system of contextual shifts which highlight certain information about an NP while backgrounding other information. See Pustejovsky [47] for discussion.

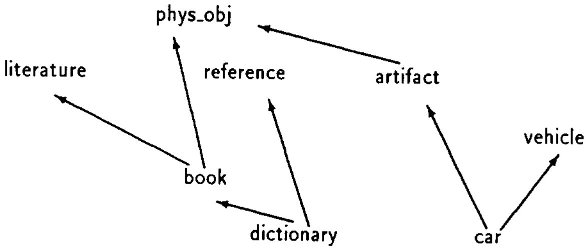

Fig. 2. Standard `is_a` hierarchy.

In this section we address, in more depth, the question of how the techniques and methods of KR relate directly to the *problem* of lexical ambiguity resolution, the *information* to bring to bear on it, and the *methods* for solving it. The discussion is carried out in the context of an alternative manifestation of the same problem, which we refer to as "hidden" lexical ambiguity. We also use the same context for presenting, informally, some aspects of the lexical inheritance structure as another level of lexical meaning.

Introducing inheritance into the lexicon is not an entirely new idea. For example, Flickinger et al. [22] discuss the value of inheritance as a representational device for capturing generalizations across classes of lexical entries. A further argument for the usefulness of inheritance mechanisms is provided by Briscoe et al. [13], who show how a mechanism of lexical inference can augment a text analysis system which performs syntactic analysis and semantic interpretation by making reference to detailed lexical decomposition of entries in the style of Pustejovsky [44].

One of the implications of positing qualia structures is the necessity to have, superimposed on the lexicon, a realization of more than one lattice structure. Earlier attempts at conceptual hierarchies faced this problem all the time: conceptual models make heavy use of multiple inheritance, as systems have to grapple with accounting for the fact that, according to particular lexical-conceptual projections, biased by a variety of context factors, different aspects of objects become more or less prominent as context varies. Thus, as illustrated in Fig. 2, a "book" `is_a` "literature", as well as a "physical_object"; a "dictionary" `is_a` "physical_object", as well as "reference"; a "car" `is_a` both "vehicle" and an "artifact", and so forth. Still, as descriptive as such relations may appear, models like these suffer from a very limited notion of lexical structure; one particular consequence of this is the ambiguity of class membership (or, in our terminology, "hidden" lexical ambiguity). Thus, even though elaborate mechanisms have been proposed to control and limit the flow of information along the generalization/specialization links, there has been no theory to either (a) explain how to assign structure to lexical items, or (b) specify lexical relations between lexical items in terms of links between only certain aspects of their respective lexical structures.

The approach presented here, with its several distinct levels of semantic description, and in particular the qualia structure, are relevant to just this issue.

On this view, a lexical item inherits information according the qualia structure it carries. In this way, the different senses for words can be rooted into suitable, but orthogonal lattices. To illustrate this point, consider the two `is_a` relations below, and the differences in what relations the objects enter into.

	play `is_a` book	dictionary `is_a` book
read	ok	no
buy	ok	ok
consult	no	ok
begin	ok (?)	no

This table illustrates a serious problem with most current inheritance systems for lexical knowledge. Namely, although it might seem reasonable to think of both plays and dictionaries as "books", they behave very differently in terms of how they are selected by different relations. This suggests that a single lattice for inheritance is inadequate for capturing the different dimensions of meaning for lexical items.

Lexical inheritance theory, on the other hand, posits a separate lattice per role in the qualia structure. Briefly, inheritance through qualia amounts to the following relations for this example:

book `is_formal` phys-object,
book `is_telic` literature,
book `is_agent` literature,
dictionary `is_formal` book,
dictionary `is_telic` reference,
dictionary `is_agent` compiled-material,
play `is_agent` literature,
play `is_telic` book.

The different inheritance structures just mentioned can be illustrated by the diagram in Fig. 3.

With the qualia roles differentiating the lattice structures, giving us a *typed inheritance*, we can exclude the unwanted inferences listed above. In the context of this paper, we assume a system of defaults operating, such as outlined in Evans and Gazdar [20] or Beierle et al. [7].

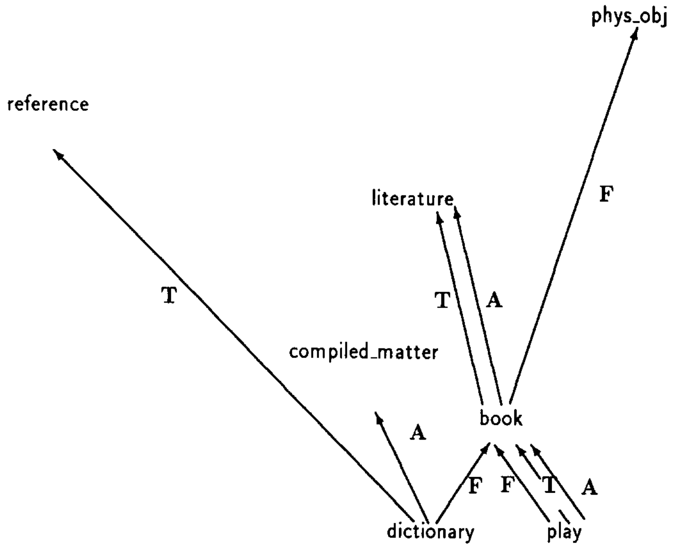

Fig. 3. Qualia-typed inheritance.

Following Touretzky [54], we will define the *fixed inheritance* structure of a lexical item as follows, where Q and P are concepts in our model of lexical organization. Then:

Definition 4.1. A sequence $< Q_1, P_1, \ldots, P_n >$ is an *inheritance path*, which can be read as the conjunction of ordered pairs $\{< x_1, y_i > \mid 1 \leqslant i \leqslant n\}$.

Furthermore, from this we can define the set of concepts that lie on an inheritance path, as distinguished by a particular quale role (*Telic* versus *Formal*). We call this the *conclusion space* for a given quale, Φ_q.

Definition 4.2. The *conclusion space* of a set of sequences, Φ_q, is the set of all pairs $< Q, P >$ such that a sequence $< Q, \ldots, P >$ appears in Φ_q, where q is one of the qualia for the concept Q.

Definition 4.3. The *complete conclusion space* Φ is the set of all conclusion spaces defined for each quale for a concept: $\Phi = \Phi_{q_i}$.

Finally, by adopting Touretzky's operator *Inh*—where, for every set of sequences S, $Inh(S)$ denotes the set of values inheritable from S, (see Pustejovsky [47] for details)—we can differentiate the lattice structures shown above for *book* as follows:

Let $[\![\alpha]\!]^{\Phi}$ stand for the denotation of α with respect to a model of inheritance over the set of sequences, Φ. Then,

$$
\begin{aligned}
&[\![book]\!]^{\Phi} = \\
&\lambda x\ [\,book(x) \wedge Formal(x) = Inh(physobj')\\
&\qquad\qquad \wedge\ Telic(x) = Inh(literature')\\
&\qquad\qquad \wedge\ Agentive(x) = Inh(literature')\,].
\end{aligned}
$$

Thus, by viewing the different facets of meaning of a lexical item (i.e. its qualia) as inheriting from orthogonally typed lattice structures, the representation will hopefully avoid many of the problems which have plagued languages with multiple inheritance, although this problem has not been fully explored within this system.[9]

5. Implications for natural language processing and knowledge representation

The method of fine-grained characterization of lexical entries, as proposed here, effectively allows us to conflate different word senses (in the traditional meaning of this term) into a single *meta-entry*, thereby offering great potential not only for systematically encoding regularities of word behavior dependent on context, but also for greatly reducing the size of the lexicon. Following Pustejovsky and Anick [48], we call such meta-entries *lexical conceptual paradigms (LCPs).* The theoretical claim here is that such a characterization constrains what a possible word meaning can be, through the mechanism of logically well-formed semantic expressions. The expressive power of a KR formalism can then be viewed as a tool which gives substance to this claim.

The notion of a meta-entry turns out to be very useful for capturing the systematic ambiguities which are so pervasive throughout language. For example, an apparently unambiguous noun such as *newspaper* can appear in many semantically distinct contexts.

(20) *The coffee cup is on top of the newspaper.*
(21) *The article is in the newspaper.*
(22) *The newspaper attacked the senator from Massachusetts.*
(23) *The newspaper is hoping to fire its editor next month.*

The noun *newspaper* falls into a particular specialization of the *Product/Producer* paradigm, where the noun can logically denote either the organization (i.e. (22) and (23)) or the product produced by the organization (i.e. (20) and (21)). This is another example of logical polysemy and is represented in the lexical structure for *newspaper*, where the qualia differentiate these aspects explicitly.

---

[9] Boguraev and Pustejovsky [11] addresses the issues relating to multiple inheritance in a more comprehensive manner.

In the previous section we examined how semantic information is inherited through typed inheritance mechanisms. What lexical conceptual paradigms illustrate very clearly, however, is that syntactic information is also inheritable between lexical items. To illustrate this point, consider the class of process/result nominals such as *merger, joint venture, consolidation*, etc. These nominals are ambiguous between an event interpretation (the *act* of merging) versus the resulting entity or state (the merger which *results*). Examples of how these nominals pattern syntactically in text are given below:[10]

(24) *A pharmaceutical joint venture of Johnson & Johnson and Merck agreed in principle to buy the U.S. over-the-counter drug business of ICI Americas for over $450 million.*
(25) *Mr. Rey brought about Titanic's merger with Society Bank of Cleveland ....*
(26) *The company announced the joint venture between its subsidiary and a Moscow cooperative to export the yarn to the Soviet Union.*
(27) *Shareholders must approve the merger meetings of the two companies in late November.*
(28) *But Mr. Rey brought about a merger between the country's major producers.*

These nominals enter into an LCP which generates a set of structural templates predicted for that noun in the language. For example, the LCP in this case is the **union** concept, and has the following lexical structure associated with it:

$$\begin{bmatrix} \textbf{union(x)} & \\ \text{CONST} = & \{\textbf{entity(y)},\textbf{entity(z)}\} \\ \text{FORMAL} = & \textbf{entity(x)} \\ \text{AGENTIVE} = & \textbf{artifact(x)} \end{bmatrix}$$

This states that a `union` is an event which brings about one entity from two or more, and is a type of artifact. The lexical structure for the nominal

[10] Among the alternations captured by LCPs are the following:
(1) Count/Mass alternations: *sheep.*
(2) Container/Containee alternations: *bottle.*
(3) Figure/Ground Reversals: *door, window.*
(4) Product/Producer diathesis: *newspaper, IBM, Ford.*
(5) Plant/Food alternations: *fig, apple.*
(6) Process/Result diathesis: *examination, combination.*
(7) Place/People diathesis: *city, New York.*
Similar alternations have been proposed for verbs as well (cf. Levin [35]). Others who have worked on regular alternations of the meanings of lexical forms include Ostler and Atkins [41] and Apresjan [4].

merger is inherited from this paradigm.

$$\begin{bmatrix} \textbf{merger(x)} & & \\ \text{CONST} & = & \{\textbf{company(y),company(z)}\} \\ \text{FORMAL} & = & \textbf{company(x)} \\ \text{AGENTIVE} & = & \textbf{artifact(x)} \end{bmatrix}$$

All synonyms for this word will share in the same LCP behavior: *merging*, *unification*, *coalition*, *combination*, *consolidation*, etc. With this LCP there are associated syntactic realization patterns for how the word and its arguments are realized in text. Such a paradigm is a very generic, domain-independent set of schemas, which is a significant advantage for multi-domain and multi-task NLP applications.

For the particular LCP of `union`, the syntactic schemas include those listed in Table 1. There are several things to note here. First, such paradigmatic behavior is extremely regular for nouns in a language, and as a result, the members of such paradigms can be found using knowledge acquisition techniques from large corpora (see for example, Pustejovsky et al [49] for one such algorithm). Secondly, because these are very common nominal patterns for nouns such as *merger*, it is significant when the noun appears without all arguments explicitly expressed. For example, in (29) below, presuppositions from the lexical structure combine with discourse clues in the form of definite reference in the NP (*the merger*) to suggest that the other partner in the merger was mentioned previously in the text.

(29) *Florida National said yesterday that it remains committed to the merger.*

Similarly powerful inferences can be made from an indefinite nominal when introduced into the discourse as in (30). Here, there is a strong presupposition that both partners in the merger are mentioned someplace in the immediately local context, as a coordinate subject, since the NP is a newly mentioned entity.

Table 1

LCP schemas: [where N=UNION; X=arg1; Y=arg2]	EXAMPLE
N of X and Y	merger of x and y (24)
X's N with Y	x's merger with y (25)
Y's N with X	y's merger with x (25)
N between X and Y	merger between x and y (26)
N of Z (Z = X + Y)	merger of the two companies (27)
N between Z	merger between two companies (28)

(30) *Orkem and Coates said last Wednesday that the two were considering a merger, through Orkem's British subsidiary, Orkem Coatings U.K. Ltd.*

Thus, the lexical structures provide a rich set of schemas for argument mapping and semantic inferencing, as well as directed presuppositions for discontinuous semantic relations.

One final and important note about lexical structures and paradigmatic behavior. The seed information for these structures is largely derivable from machine-readable dictionaries. For example, a dictionary definition for *merger* (from the *Longman Dictionary of Contemporary English*) is "the joining of 2 or more companies or firms" with subject code FINANCE. This makes the task of automatic construction of a robust lexicon for NLP applications a very realizable goal (cf. Boguraev [9] and Wilks et al. [58]).

Although we have concentrated in this paper on how particular KR formalisms can contribute to lexical semantic modeling, there are clearly implications for KR in adopting the theory and methodology outlined above. By linking semantic representations to observable patterns of linguistic behavior in different languages, one can avoid *ad hoc* constructs and assumptions about the objects being modeled. One can conclude, for example, on the basis of our previous discussion, that nouns such as *window*, *door*, and *room* are conceptually relations between a physical object and an aperture or enclosure of some sort. We can infer this directly from the patterns of polysemy in the language, where the noun is ambiguous between either sense. Contrast the behavior of this class with nouns such as *lid*, *cap*, and *cover*, which can only refer to the physical object, and not the aperture. Conceptually, then, we conclude that these nouns do not act in such a relational capacity.

This methodology extends generally to all language categories, and therefore to all concept types in the knowledge representation domain. One particular consequence of the analyses developed in this paper, initially made possible from methods in knowledge representation, is that they help further define the boundary between lexical and commonsense knowledge. Typically, vastly different types of knowledge and knowledge sources are made available, via a KR formalism, to enable a variety of inferential processes. What tends to get encoded in a knowledge base is, by general admission, commonsense knowledge; to date, very little attention has been paid to separating information of different types. There are, however, clear advantages to factoring out the linguistic facts, and not mixing the domains of lexical inference and commonsense reasoning.

What we have attempted to do in this paper is to look at linguistic and syntactic generalizations, and to explain the systematic patterning and behavior of words involving semantic distinctions by introducing well-defined semantic mechanisms. By virtue of being more predictable and better behaved than general commonsense mechanisms, these generalizations highlight the *de facto* differences between linguistic and commonsense knowledge and the inferencing mechanisms associated with them.

6. Conclusion

We have outlined a framework for lexical semantic research that we believe can be useful for both computational linguists and theoretical linguists alike. We argued against the view that word meanings are fixed and inflexible, where lexical ambiguity must be treated by multiple word entries in the lexicon. Rather, the lexicon can be seen as a generative system, where word senses are related by logical operations defined by the well-formedness rules of the semantics. In this view, much of the lexical ambiguity of highly ambiguous lexical items is explained because the semantic load is spread more evenly throughout the lexicon to the other lexical categories; furthermore, the lexical knowledge we propose as necessary for ambiguity resolution is seen as factored out at different levels of lexical representation. We looked at two of these levels, qualia structure and lexical inheritance, as they turn out to be of particular relevance to the structuring of the semantic information carried by nouns and adjectives, and applying it, via composition, to the construction of semantic interpretation of complex expressions. The methods underlying the analysis of ambiguous phrases and the construction of corresponding semantic expressions make extended use of KR devices and techniques. A computational realization of this lexical semantics makes the problem of lexical ambiguity resolution more tractable. Furthermore, because a word's paradigmatic syntactic behavior follows largely from its being identified with a certain semantic type, the approach we have outlined also improves the overall robustness of automatic natural language processing—both in terms of lexicon acquisition and language learnability.

Acknowledgements

We would like to thank Sue Atkins, Ted Briscoe, Bob Ingria, and Scott Waterman for many useful discussions and feedback on earlier versions of this paper. We would like to especially thank Barbara Grosz, Fernando Pereira, and an anonymous reviewer whose careful readings and comments helped to significantly improve the final text.

References

[1] H. Aït-Kaci, A lattice-theoretic approach to computation based on a calculus of partially ordered types, Ph.D. Thesis, University of Pennsylvania, Philadelphia, PA (1984).

[2] J. Allen, Maintaining knowledge about temporal intervals, *Commun. ACM* **26** (1983) 832–843.

[3] P. Anick and J. Pustejovsky, An application of lexical semantics to knowledge acquisition from corpora, in: *Proceedings Thirteenth International Conference on Computational Linguistics*, Helsinki, Finland (1990) 7–12.

[4] J. Apresjan, Regular polysemy, *Linguistics* **142** (1973).

[5] B. Atkins, Building a lexicon: reconciling anisomorphic sense differentiations in machine-readable dictionaries, Presented at "Natural Language in the 90's—Language and Action in the World", Cambridge, MA (1990).

[6] B. Atkins, J. Kegl and B. Levin, Anatomy of a verb entry: from linguistic theory to lexicographic practice, *Int. J. Lexicography* **1** (1988) 84–126.

[7] C. Beierle, U. Hedtstück, U. Pletat, P.H. Schmitt and J. Siekmann, An order-sorted logic for knowledge representation systems, *Artif. Intell.* **55** (1992) 149–191.

[8] M. Bierwisch, Semantische und konzeptuelle Repräsentationen lexikalischer Einheiten, in: R. Ružička and W. Motsch, eds., *Untersuchungen zur Semantik* (Akademische-Verlag, Berlin, 1983).

[9] B. Boguraev, Building a lexicon: the contribution of computational lexicography, in: L. Bates and R. Weischedel, eds., *Challenges in Natural Language Processing* (Cambridge University Press, Cambridge, 1992).

[10] B. Boguraev and E. Briscoe, *Computational Lexicography for Natural Language Processing* (Longman, London, 1989).

[11] B. Boguraev and J. Pustejovsky, *Lexical Knowledge: Representation and Acquistion* (MIT Press/Bradford Books, Cambridge, MA, to appear).

[12] R.J. Brachman and J. Schmolze, An overview of the KL-ONE knowledge representation system, *Cogn. Sci.* **9** (1985) 171–216.

[13] E. Briscoe, A. Copestake and B. Boguraev, Enjoy the paper: lexical semantics via lexicology, in: *Proceedings Thirteenth International Conference on Computational Linguistics*, Helsinki, Finland (1990) 42–47.

[14] N. Calzolari, Acquiring and representing semantic information in a lexical knowledge base, in: J. Pustejovsky and S. Bergler, eds., *Lexical Semantics and Knowledge Representation* (Springer, New York, 1992).

[15] J.G. Carbonell, Default reasoning and inheritance mechanisms on type hierarchies, in: *Proceedings Workshop on Data Abstraction, Databases, and Conceptual Modelling*, Pingree Park, CO (1980).

[16] L. Cardelli and P. Wegner, On understanding types, data abstraction, and polymorphism, *ACM Comput. Surv.* **17** (1985) 471–522.

[17] A. Copestake, The ACQUILEX LKB: representation issues in semi-automatic acquisition of large lexicons, in: *Proceedings Third Conference on Applied Natural Language Processing*, Trento, Italy (1992) 88–95.

[18] A. Copestake and E. Briscoe, Lexical operations in a unification-based framework, in: J. Pustejovsky and S. Bergler, eds., *Lexical Semantics and Knowledge Representation* (Springer, New York, 1992).

[19] W. Croft, Categories and relations in syntax: the clause-level organization of information, Ph.D. Dissertation, Stanford University, Stanford, CA (1986).

[20] R. Evans and G. Gazdar, Inference in DATR, in: *Proceedings Fourth Conference of the European Association for Computational Linguistics*, Manchester, England (1989).

[21] C. Fillmore, Frames and the semantics of understanding, *Q. Semantica* **6** (1985) 222–254.

[22] D. Flickinger, C. Pollard and T. Wasow, Structure-sharing in lexical representation, in: *Proceedings 23rd Conference of the European Association for Computational Linguistics*, Chicago, IL (1985) 262–267.

[23] J. Grimshaw, Complement selection and the lexicon, *Ling. Inquiry* **10** (1979) 279–326.

[24] J. Grimshaw, *Argument Structure* (MIT Press, Cambridge, 1990).

[25] P.J. Hayes, Naive physics manifesto, in: J.R. Hobbs and R.C. Moore, eds., *Formal Theories of the Commonsense World* (Ablex, Norwood, NJ, 1984).

[26] A. Herskovits, *Language and Spatial Cognition* (Cambridge University Press, Cambridge, 1986).

[27] G.E. Hirst, *Semantic Interpretation and the Resolution of Ambiguity* (Cambridge University Press, Cambridge, 1987).

[28] J.R. Hobbs, W. Croft, T. Davies, D. Edwards and K. Laws, Commonsense metaphysics and lexical semantics, *Comput. Linguistics* **13** (1987) 241–250.

[29] R. Jackendoff, Multiple subcategorizations and the θ-criterion: the case of *Climb*, *Natural Language Linguistic Theor.* **3** (1985) 271–295.

[30] J. Katz, Semantic theory and the meaning of 'good', *J. Philos.* **61** (1964) 739–766.

[31] P. Kiparsky and C. Kiparsky, Fact, in: M. Bierwisch and K. Heidolph, eds., *Progress in Linguistics* (Mouton, The Hague, 1970) 143–173.

[32] E. Klein and J. van Benthem, Categories, polymorphism and unification, Centre for Cognitive Science, University of Edinburgh, and Institute for Language, Logic and Information, University of Amsterdam (1987).

[33] G. Lakoff, *Women, Fire, and Dangerous Things* (The University of Chicago Press, Chicago, IL, 1987).

[34] G. Lakoff and M. Johnson, *Metaphors We Live By* (The University of Chicago Press, Chicago, IL, 1980).

[35] B. Levin, *Towards a Lexical Organization of English Verbs* (University of Chicago Press, Chicago, IL, to appear).

[36] I. Mel'čuk, Semantic description of lexical units in an explanatory combinatorial dictionary, *Int. J. Lexicography* **1** (1988).

[37] M. Moens and M. Steedman, Temporal ontology and temporal reference, *Comput. Linguistics* **14** (1988) 15–28.

[38] M. Moens, J. Calder, E. Klein, M. Reape and H. Zeevat, Expressing generalizations in unification-based grammar formalisms, in: *Proceedings Fourth Conference of the European Association for Computational Linguistics*, Manchester, England (1989) 174–181.

[39] R. Montague, The proper treatment of quantification in ordinary English, in: K.J.J. Hintikka, J.M. Moravcsik and P.C. Suppes, eds., *Approaches to Natural Language: Proceedings of the 1970 Stanford Workshop on Grammar and Semantics* (Reidel, Dordrecht, Netherlands, 1973) 221–242.

[40] J.M. Moravcsik, Aita as generative factor in Aristotle's philosophy, *Dialogue* (1975).

[41] N. Ostler and B.T.S. Atkins, Predictable meaning shift: some linguistic properties of lexical implications rules, in: J. Pustejovsky and S. Bergler, eds., *Lexical Semantics and Knowledge Representation* (Springer, New York, 1992).

[42] R. Passonneau, A computational model of the semantics of tense and aspect, *Comput. Linguistics* **14** (1988).

[43] C. Pollard and I. Sag, *Information-Based Syntax and Semantics*, CSLI Lecture Notes **13** (Stanford University Press, Stanford, CA 1987).

[44] J. Pustejovsky, The generative lexicon, *Comput. Linguistics* **17** (1991) 409–441.

[45] J. Pustejovsky, The syntax of event structure, *Cognition* **41** (1991) 47–81.

[46] J. Pustejovsky, Type coercion and lexical inheritance, in: J. Pustejovsky, ed., *Semantics and the Lexicon* (Kluwer Academic Publishers, Dordrecht, Netherlands, 1993).

[47] J. Pustejovsky, *The Generative Lexicon: A Theory of Computational Lexical Semantics* (MIT Press, Cambridge, MA, to appear).

[48] J. Pustejovsky and P. Anick, On the semantic interpretation of nominals, in: *Proceedings Twelfth International Conference on Computational Linguistics*, Budapest, Hungary (1988) 518–523.

[49] J. Pustejovsky, S. Bergler and P. Anick, Lexical semantic techniques for corpus analysis, *Comput. Linguistics* (to appear); Special Issue on Corpus Linguistics.

[50] M. Quillian, Semantic memory, in: M. Minsky, ed., *Semantic Information Processing* (MIT Press, Cambridge, MA, 1968).

[51] R. Schank, *Conceptual Information Processing* (American Elsevier, New York, 1975).

[52] S. Shieber, *An Introduction to Unification-Based Approaches to Grammar* CSLI Lecture Notes **4** (Stanford University Press, Stanford, CA, 1986).

[53] L. Talmy, Lexicalization Patterns, in: T. Shopen, ed., *Language Typology and Syntactic Description* (Cambridge University Press, Cambridge, 1985).

[54] D.S. Touretzky, *The Mathematics of Inheritance Systems* (Morgan Kaufmann, Los Altos, CA, 1986).

[55] U. Weinreich, *Explorations in Semantic Theory* (Mouton, The Hague, 1972).

[56] Y. Wilks, An intelligent analyzer and understander for English, *Commun. ACM* **18** (1975) 264–274.

[57] Y. Wilks, Making preferences more active, *Artif. Intell.* **10** (1978) 75–97.

[58] Y. Wilks, D. Fass, C.-M. Guo, J. McDonald, T. Plate, B. Slator, A tractable machine dictionary as a resource for computational semantics, in: B. Boguraev and E. Briscoe, eds., *Computational Lexicography for Natural Language Processing* (Longman, London, 1989) 193–228.

[59] W. Woods, What's in a link: foundations of semantics networks, in: D.G. Bobrow and A. Collins, eds., *Representation and Understanding: Studies in Cognitive Science* (Academic Press, New York, 1975).