

A FOUNDATION FOR SEMANTIC INTERPRETATION

Graeme Hirst
Department of Computer Science
Brown University
Providence, RI 02912

Abstract

Traditionally, translation from the parse tree representing a sentence to a semantic representation (such as frames or procedural semantics) has always been the most ad hoc part of natural language understanding (NLU) systems. However, recent advances in linguistics, most notably the system of formal semantics known as Montague semantics, suggest ways of putting NLU semantics onto a cleaner and firmer foundation. We are using a Montague-inspired approach to semantics in an integrated NLU and problem-solving system that we are building. Like Montague's, our semantics are compositional by design and strongly typed, with semantic rules in one-to-one correspondence with the meaning-affecting rules of a Marcus-style parser. We have replaced Montague's semantic objects, functors and truth conditions, with the elements of the frame language Frail, and added a word sense and case slot disambiguation system. The result is a foundation for semantic interpretation that we believe to be superior to previous approaches.

1. Introduction

By *semantic interpretation* we mean the process of mapping from a syntactically analyzed sentence of natural language to a representation of its meaning. We exclude from semantic interpretation any consideration of discourse pragmatics; rather, discourse pragmatics operate upon the output of the semantic interpreter. We also exclude syntactic analysis; the integration of syntactic and semantic analysis becomes very messy when complex syntactic constructions are considered, and, moreover, it is our observation that those who argue for the integration of the two are usually arguing for subordinating the role of syntax, a position we reject. This is not to say that parsing can get by without semantic help; indirect object finding,

This work was supported by the Office of Naval Research under contract number N00014-79-C-0592.

and prepositional phrase and relative clause attachment, for example, often require semantic knowledge. Below we will show that syntax and semantics may work well together while remaining distinct modules.

Research on semantic interpretation in artificial intelligence goes back to Woods's dissertation (1967, 1968), which introduced procedural semantics in a natural-language front-end for an airline reservation system. Woods's system had rules with patterns that, when they matched part of the parsed input sentence, contributed a string to the semantic representation of the sentence. This string was usually constructed from the terminals of the matched parse tree fragment. The strings were combined to form a procedure call that, when evaluated, entered or retrieved the appropriate database information. This approach is still the predominant one today, and even though it has been refined over the years, semantic interpretation remains perhaps the least understood and most ad hoc area of natural language understanding (NLU).¹

However, recent advances in linguistics, most notably Montague semantics (Montague 1973; Dowty, Wall and Peters 1981), suggest ways of putting NLU semantic interpretation on a cleaner and firmer foundation than it now is. In this paper, we describe such a foundation.²

2. Montague semantics

In his well-known "PTQ" paper (Montague 1973), Richard Montague presented the complete syntax and semantics for a small fragment of English. Although it was limited in vocabulary and syntactic complexity, Montague's fragment dealt with such impor-

¹It is also philosophically controversial. For discussion, see Fodor 1978, Johnson-Laird 1978, Fodor 1979, and Wilks 1982.

²Ours is not the only current work with this goal; in Section 7 we discuss other similarly motivated work.

tant semantic problems as opaque contexts, different types of predication with the word *be*, and the “the temperature is 90” problem;³ for details of these, see Dowty, Wall and Peters (1981).

Montague’s semantic rules correspond to what we have been calling semantic interpretation. That is, in conjunction with a syntactic process, they produce a semantic representation, or translation, of a sentence. There are four important properties of Montague semantics that we will examine here. Below, we will carry three of these properties over into our own semantics.

The first property, the one that we will later drop, is that for Montague, semantic objects, the results of the semantic translation, were such things as *individual concepts* (which are functions to individuals from the cartesian product of points in time and possible worlds), properties of individual concepts, and functions of functions of functions of functions. At the top level, the meaning of a sentence was a truth condition relative to a possible world and point in time. These semantic objects were represented by expressions of *intensional logic*; that is, instead of translating English directly into these objects, a sentence was first translated to an expression of intensional logic, for which, in turn, there existed an interpretation in terms of these semantic objects.

Second, Montague had a strong theory of types for his semantic objects: a set of types that corresponded to types of syntactic constituents. Thus, given a particular syntactic category, such as proper noun or adverb, Montague was able to say that the meaning of a constituent of that category was a semantic object of such and such a type.⁴ Montague’s system of types was recursively defined, with entities, truth values and intensions as primitives, and other types defined as functions from one type to another in such a manner that if syntactic category *X* was formed by adding category *Y* to category *Z*, then the type corresponding to *Z* would be functions from senses of the type of

³That is, to ensure that “The temperature is 90 and the temperature is rising” cannot lead to the inference that “90 is rising”.

⁴To be precise: the semantic type of a proper noun is set of properties of individual concepts; that of an adverb is function between sets of individual concepts (Dowty et al 1981: 183, 187).

Y to the type of *X*.⁵

Third, in Montague’s system the syntactic rules and semantic rules are in one-to-one correspondence. Each time a particular syntactic rule applies, so does the corresponding semantic rule; while the one operates on some syntactic elements to create a new element, the other operates on the corresponding semantic objects to create a new object that will correspond to the new syntactic element. Thus the two sets of rules operate in tandem.

Fourth, Montague’s semantics is *compositional*, which is to say that the meaning of the whole is a systematic function of the meaning of the parts. At first glance this sounds trivial; if the noun phrase *my pet penguin* denotes by itself some particular entity, namely the one sitting on my lap as I write this paper, then we do not expect it to refer to a different entity when it is embedded in the sentence *I love my pet penguin*, and a semantic system that did not reflect this would be a loser indeed. Yet there are alternatives to compositional semantics.

The first alternative is that the meaning of the whole is a function of not just the parts but also the situation in which the sentence is uttered. For example, the possessive in English is highly dependent upon pragmatics; the phrase *Nadia’s penguin* could refer, in different circumstances, to the penguin that Nadia owns, to the one that she is carrying but doesn’t actually own, or to the one that she just bet on at the penguin races. Our definition above of semantic interpretation excluded this sort of consideration, but this should not be regarded as uncontroversial.

The second alternative to compositional semantics is that the meaning of the whole is not a *systematic* function of the parts in any reasonable sense of the word. This is exemplified by the interpretation of the word *depart* in Woods’s original system, which varied greatly depending on the preposition it dominated (Woods 1967:A-43–A-46). For example, the interpretation of the sentence:

AA-57 departs from Boston.

is, not unreasonably:

⁵For example, the semantic type of prepositions is functions mapping senses of the type of noun phrases to the semantic type of prepositional phrases.

depart (aa-57, boston).

That is, the semantic object into which *depart* is translated is the procedure *depart*. (AA-57 is an airline flight.) However, the addition of a prepositional phrase changes this; Table 1 shows the interpretation of the same sentence after various prepositional phrases have been appended. For example, the addition of *to Chicago* changes the translation of *depart* to *connect*, though the intended sense of the word is clearly unchanged.⁶

This is necessitated by the particular set of database primitives that Woods used, selected for their being "atomic" (1967:7-4-7-11) rather than for promoting compositionality. Rules in the system are able to generate non-compositional representations because they have the power to set an arbitrarily complex parse tree as their trigger, and to return an arbitrary representation that could modify or completely ignore the components of the parse trees they are supposed to be interpreting.⁷ For example, a rule can say (1967:A-44):

If you have a sentence whose subject is a flight, whose verb is *leave* or *depart*, and which has two (or more) prepositional phrases modifying the verb, one with *from* and a place name, the other with *at* and a time, then the interpretation is *equal* (*dtime* (a, b), c), where a is the flight, b is the place, and c is the time.

Thus while Woods's semantics could probably be made reasonably compositional simply by appropriate adjustment of the procedure calls into which sentences are translated, it would still not be compositional by design the way Montague semantics is.

⁶We have simplified a little here in order to make our point. In fact, sentences like those in Table 1 with prepositional phrases will actually cause the execution of two semantic rules: one for the complete sentence, and one for the sentence it happens to contain, *AA-57 departs from Boston*. The resulting interpretation will be the conjunction of the output from each rule (Woods 1967:9-5):

AA-57 departs from Boston to Chicago.
depart (aa-57, boston) and *connect* (aa-57, boston, chicago)
Woods leaves it open (1967:9-7) as to how the semantic redundancy in such expressions should be handled, though one of his suggestions is a filter that would remove conjuncts implied by others, giving, in this case, the interpretation shown in Table 1.

⁷Nor is there anything that prevents the construction of rules that would result in conjunctions with conflicting, rather than merely redundant, terms.

TABLE 1.

NONCOMPOSITIONALITY IN WOODS'S SYSTEM

AA-57 departs from Boston.

depart (aa-57, boston)

AA-57 departs from Boston to Chicago.

connect (aa-57, boston, chicago)

AA-57 departs from Boston on Monday.

dday (aa-57, boston, monday)

AA-57 departs from Boston at 8:00am.

equal (*dtime* (aa-57, boston), 8:00am)

AA-57 departs from Boston after 8:00am.

greater (*dtime* (aa-57, boston), 8:00am)

AA-57 departs from Boston before 8:00am.

greater (8:00am, *dtime* (aa-57, boston))

Although Montague semantics has much to recommend it, it is not possible, however, to implement it directly in a practical NLU system, for two reasons. The first is that Montague semantics as currently formulated is computationally impractical. It throws around huge sets, infinite objects, functions of functions, and piles of possible worlds with great abandon. Friedman, Moran and Warren (1978a) point out that in the smallest possible Montague system, one with two entities and two points of reference, there are, for example, $2^{2^{22}}$ elements in the class of possible denotations of prepositions, each element being a set containing 2^{512} ordered pairs.⁸

The second reason we can't use Montague semantics directly is that truth-conditional semantics are not useful in AI; AI uses *knowledge semantics* (Tarnawsky 1982) in which semantic objects tend to be symbols or expressions in a declarative or procedural knowledge representation system. Moreover, truth-conditional semantics really only deals with declarative sentences (Dowty et al 1981:13) (though there has been work attempting to extend Montague's work to questions; e.g. Hamblin 1973); a practical NLU system needs to be able to deal with commands and questions as well as declarative sentences.

⁸Despite this problem, Friedman et al (1978b, 1978c) have implemented Montague semantics computationally by using techniques for maintaining partially specified models. However, their system is intended as a tool for understanding Montague semantics better, rather than as a usable NLU system (1978b:28).

There have, however, been attempts to take the intensional logic that Montague uses as an intermediate step in his translations, and give it a new interpretation in terms of AI-type semantic objects, thus preserving all other aspects of Montague's approach; see, for example, Hobbs and Rosenschein 1977, and Smith's (1979) objections to their approach. There has also been interest in using the intensional logic itself (or something similar) as an AI representation⁹ (e.g. Moore 1981). But while it may be possible to make limited use of intensional logic expressions,¹⁰ there are many problems that need to be solved before intensional logic or other flavors of logical forms could support the type of inference and problem solving that AI requires of its semantic representations; see Moore 1981 for a useful discussion. Moreover, Gallin (1975) has shown Montague's intensional logic to be incomplete. (See also the discussion in Section 7 of work using logical forms.)

Nevertheless, it is possible to use many aspects of Montague's approach in semantics in AI. The semantic interpreter that we describe below maintains three of the four properties of Montague semantics that we described above, and we therefore refer to it as "Montague-inspired".

3. Our semantic interpreter

Our semantic interpreter is a component of a system that uses a frame-like representation for both story comprehension and problem-solving. The system includes a frame language, named Frail, a problem solver, and a discourse pragmatics component; further details may be found in Charniak 1981, Wong 1981a, and Wong 1981b. The natural language front-end includes Paragram, a deterministic parser based on that of Marcus (1980). Unlike Marcus's parser, Paragram has two types of rule: base phrase structure rules and transformational rules. It is also able to parse ungrammatical sentences; it always uses the rule that matches best, even if none match exactly. Paragram is described in Charniak 1983.

⁹Ironically, Montague regarded intensional logic merely as a convenience in specifying his translation, and one that was completely irrelevant to the substance of his semantic theories.

¹⁰Godden (1981) in fact uses them for simple translation between Thai and English.

TABLE 2.

TYPES IN THE ABSITY SEMANTIC INTERPRETER

BASIC TYPES

Frame^a
 (penguin ?x), (love ?x)

Slot
 color, agent

Frame determiner^b
 (the ?x), (a ?x)

OTHER TYPES

Slot-filler pair = slot + frame statement
 (color=red), (agent=(the ?x (fish ?x)))

Frame descriptor = frame + slot-filler pair*
 (penguin ?x (owner=Nadia)),
 (love ?x (agent=Ross) (patient=Nadia)),
 (dog ?x)

Frame statement [or instance^c]
 = frame determiner + frame descriptor
 (the ?x (penguin ?x (owner=Nadia))),
 (a ?x (love ?x (agent=Ross)
 (patient=Nadia))),
 (the ?x (dog ?x)),
 penguin87 [an instance]

^aThe question-mark prefix indicates a variable. Whenever a free variable in a frame is bound to a variable in a frame determiner, a unique new name is generated for that variable and its bindings. In this paper, we shall assume for simplicity that variable names are magically "correct" from the start.

^bDo not be misled by the fact that frames and frame determiners look similar. They are actually very different: the first is a static data structure; the second is a frame retrieval procedure.

^cAn instance is the result of evaluating a frame statement in Frail. It is a symbol that denotes the object referenced by the frame statement. To Absity, there is no distinction between the two; an instance can be used wherever a frame statement can.

The semantic interpreter is named Absity (for reasons too obscure to burden the reader with). As we mentioned above, it retains three of the four properties of Montague semantics that we discussed. The property that we have dropped is, of course, truth conditionality and Montague's associated treasury of semantic objects. We have replaced them with AI-style semantics, and our own repertory of objects,

TABLE 3.
TYPE CORRESPONDENCES IN ABSITY

SYNTACTIC TYPE	SEMANTIC TYPE
Major sentence	Frame statement, instance
Sentence	Frame descriptor
Noun	Frame
Adjective	Slot-filler pair
Determiner	Frame determiner
Noun phrase	Frame statement, instance
Preposition	Slot name
Prepositional Phrase	Slot-filler pair
Verb	(Action) frame
Adverb	Slot-filler pair
Auxiliary	Slot-filler pair
Verb phrase	Frame descriptor
Clause end	Frame determiner .

which are components of the frame language Frail.¹¹

We do, however, retain a strong typing upon our semantic objects, that is, each syntactic category has an associated semantic type. Table 2 shows the types of components of Frail, how they may be combined, and examples of each; the nature of the components listed will become clearer with the examples in the next section. Table 3 gives the component of Frail that corresponds to each syntactic type. As a consequence of the kind of semantic objects we are dealing with, the system of types is not recursively defined in the Montague style, but we retain the idea that the type of a semantic object should be a function of the types of the components of that object.

We have also carried over from Montague semantics the operation of syntactic and semantic rules in tandem upon corresponding objects. However, it is not possible to maintain the one-to-one correspondence of rules when we replace Montague's simple syntax with the much larger English grammar of the Paragram parser. This is because in Montague's system each syntactic rule either creates a new node from old ones—for example, forming an intransitive verb phrase from a transitive verb and a noun phrase—or places a new

¹¹ Although the object that represents a sentence is a procedure call in Frail upon a knowledge base, this is not procedural semantics in the strict Woods sense, as the meaning inheres not in the procedures but in the objects they manipulate.

node under an existing one—such as adding an adverb to an existing intransitive verb phrase. These are actions that clearly have semantic counterparts. When we start to add movement rules such as passivization and dative movement to the grammar, we find ourselves with rules that have no clear semantic counterpart; indeed with rules that, it is often claimed (e.g. Chomsky 1965:132), leave the meaning of a sentence quite unchanged.

We therefore distinguish between parser rules that should have corresponding semantic rules and those that should not. As the above discussion suggests, rules that attach nodes are the ones that have semantic counterparts. In Paragram, these are the base structure rules. For this subset of the syntactic rules, semantic rules run in tandem, just as in Montague's semantics.¹²

It is a consequence of the above properties of our semantic interpreter that we have also retained the property of compositionality by design. This follows from the uniform typing; the correspondence between syntactic and semantic rules that maintains this uniformity; and there being a unique semantic object corresponding to each word of English¹³ (see Dowty *et al* 1981:180–181). Unlike those of Woods's (1967) airline reservation system front-end discussed in Section 2, our semantic rules are very weak: they cannot change or ignore the components upon which they operate, nor can more than one rule volunteer an interpretation for any node of the parse tree. The power of the system comes from the nature of the semantic objects and the syntax-directed application of semantic rules, rather than from the semantic rules themselves.

4. Examples

Some examples will make our semantic interpreter clearer. First, let's consider a simple noun phrase, *the book*. From Table 3, the semantic type for the determiner *the* is a frame determiner function, in this case (*the ?x*), and the type for the noun *book* is a kind of frame, here (*book ?x*). These are combined

¹² In her synthesis of transformational syntax with Montague semantics, Partee (1973, 1975) observes that the semantic rule corresponding to many transformations will simply be the identity mapping.

¹³ We show in Section 6 how this may be reconciled with lexical ambiguity.

in the canonical way—the frame name is added as an argument to the frame determiner function—and the result, (the ?x (book ?x)), is a Frail frame statement (which evaluates to an instance) that represents the unique book referred to.¹⁴

A descriptive adjective corresponds to a slot-filler pair; for example, *red* is represented by (color=*red*), where *color* is the name of a slot and *red* is a frame instance, the name of a frame. A slot-filler pair can be added as an argument to a frame, so *the red book* would have the semantic interpretation (the ?x (book ?x (color=*red*))).

Now let's consider a complete sentence:

Nadia bought the book from a store in the mall.

Table 4 shows the representation for each component of the sentence; note that the basic noun phrases have already been formed in the manner described above. Note also that we have inserted the pseudo-prepositional subject and object markers *SUBJ* and *OBJ*, which are then treated as if they were real prepositions; see Hirst and Charniak 1982 or Hirst 1983 for details of this. For simplicity, we assume that each word is unambiguous (we discuss our disambiguation procedures in Section 6); we also ignore the tense on the verb. Table 5 shows the next four stages in the interpretation. First, noun phrases and their prepositions are combined, forming slot-filler pairs. Then the prepositional phrase *in the mall* can be attached to a *store* (since a noun phrase, being a frame, can have a slot-filler pair added to it), and the prepositional phrase *from a store in the mall* is formed. The third stage shown in the Table is the attachment of the slot-filler pairs for the three top-level prepositional phrases to the frame representing the verb. Finally, the period, which is translated as a frame determiner function, causes instantiation of the buy frame, and the translation is complete.

5. Semantic help for the parser

As we mentioned earlier, any parser will occasionally need semantic help. In Marcus-type parsers, this need occurs in rules that have the form "If semantics prefers

¹⁴Note that it is the responsibility of the frame system to determine with the help of the pragmatics module which one of the books that it may know about is the correct one in context.

TABLE 4.

ABSITY EXAMPLE

WORD OR PHRASE	SEMANTIC OBJECT
<i>SUBJ</i>	<i>agent</i>
<i>Nadia</i>	(the ?x (thing ?x (propername="Nadia")))
<i>bought</i>	(buy ?x)
<i>OBJ</i>	<i>patient</i>
<i>the book</i>	(the ?y (book ?y))
<i>from</i>	<i>source</i>
<i>a store</i>	(a ?z (store ?z))
<i>in</i>	<i>location</i>
<i>the mall</i>	(the ?w (mall ?w))
<i>. [period]</i>	(a ?u)

X over Y then do X'; otherwise do Y''. To answer such questions, we have a *Semantic Enquiry Desk* that operates upon the same semantic objects as the semantic interpreter. Because these objects are components of the Frail frame language, the Enquiry Desk can use the full retrieval and inference power of Frail in answering the enquiry.

6. Word sense disambiguation

One problem that Montague semantics does not address is that of word disambiguation. Rather, there is assumed to exist a function that maps each word to a unique sense, and the semantic formalism operates on the values of this function.¹⁵ Clearly, however, a practical NLU system must take account of word sense ambiguity, and so we must add a disambiguation facility to our interpreter. Fortunately, the word translation function allows us to add this facility transparently. Instead of simply mapping a word to an invariant unique sense, the function can map it to whatever sense is correct for a particular instance.

Our disambiguation facility is called *Polaroid Words*.¹⁶ Each word in the system is represented by

¹⁵This is not quite true. Specified unique translations are given for proper names and for a few important function words, such as *the* and *be*; see Montague 1973[2]:261, or Dowty et al 1981:192ff.

¹⁶*Polaroid* is a trademark of the Polaroid Corporation.

TABLE 5.

ABSITY EXAMPLE (CONTINUED)

SUBJ Nadia
 (agent=(the ?x
 (thing ?x (propername="Nadia"))))

OBJ the book
 (patient=(the ?y (book ?y)))

in the mall
 (location=(the ?w (mall ?w)))

a store in the mall
 (a ?z (store ?z
 (location=(the ?w (mall ?w)))))

from a store in the mall
 (source=(a ?z (store ?z
 (location=(the ?w (mall ?w)))))

Nadia bought the book from a store in the mall
 (buy ?u
 (agent=(the ?x (thing ?x
 (propername="Nadia"))))
 (patient=(the ?y (book ?y)))
 (source=(a ?z (store ?z
 (location=(the ?w (mall ?w)))))

Nadia bought the book from a store in the mall.
 (a ?u
 (buy ?u
 (agent=(the ?x (thing ?x
 (propername="Nadia"))))
 (patient=(the ?y (book ?y)))
 (source=(a ?z (store ?z
 (location=(the ?w (mall ?w)))))

a separate process that, by talking to other processes and by looking at paths made by spreading activation in the knowledge base, figures out the word's meaning. Each word is like a self-developing photograph that can be manipulated by the semantic interpreter even while the picture is forming; and if some other process needs to look at the picture (e.g. if the Semantic Enquiry Desk has an "if semantics prefers" question from the parser), then a half-developed picture may provide enough information. Exactly the same process, without the spreading-activation phase,

is used to disambiguate case roles as well. Polaroid Words are described more fully in Hirst and Charniak 1982 and Hirst 1983.

7. Comparison with other work

Our approach to semantic interpretation may usefully be compared with other recent work with similar goals to ours.

One such project is that of Jones and Warren (1982), who attempt a conciliation between Montague semantics and a conceptual dependency representation (Schank 1975). Their approach is to modify Montague's translation from English to intensional logic so that the resulting expressions have a canonical interpretation in conceptual dependency. They do not address such issues as extending Montague's syntax, nor whether their approach can be extended to deal with more modern Schankian representations (e.g. Schank 1982). Nevertheless, their work, which they describe as a hesitant first step, is similar in spirit to ours, and it will be interesting to see how it develops.

Important recent work that extends the syntactic complexity of Montague's work is that on generalized phrase structure grammar (GPSG) (Gazdar 1982). Such grammars combine a complex transformation-free syntax with Montague's semantics, the rules again operating in tandem. Gawron *et al* (1982) have implemented a database interface based on GPSG. In their system, the intensional logic of the semantic component is replaced by a simplified extensional logic, which, in turn, is translated into a query for database access. Schubert and Pelletier (1982) have also sought to simplify the semantic output of a GPSG to a more "conventional" logical form; and Rosenschein and Shieber (1982) describe a similar translation process into extensional logical forms, using a context-free grammar intended to be similar to a GPSG.¹⁷

The GPSG approaches differ from ours in that their output is a logical form rather than an immediate representation of a semantic object; that is, the output is not tied to any representation of knowledge. In Gawron *et al*'s system, the database

¹⁷ Rosenschein and Shieber's semantic translation follows parsing rather than running in parallel with it, but it is strongly syntax-directed, and is, it seems, isomorphic to an in-tandem translation that provides no feedback to the parser.

provides an interpretation of the logical form, but only in a weak sense, as the form must first pass through another (apparently somewhat ad hoc) translation and disambiguation process. Nor do these approaches provide any semantic feedback to the parser.¹⁸ These differences, however, are independent of the choice of GPSG; it should be easy, at least in principle, to modify these approaches to give Frail output, or, conversely, to replace Paragram in our system with a GPSG parser.¹⁹

The PSI-KLONE system of Bobrow and Webber (1980a, 1980b) also has a close coupling between syntax and semantics. Rather than operating in tandem, though, the two are described as “cascaded”, with an ATN parser handing constituents to a semantic interpreter, which is allowed to return them (causing the ATN to back up) if the parser’s choice is found to be semantically untenable. Otherwise, a process of *incremental description refinement* is used to interpret the constituent; this relies on the fact that the syntactic constituents are represented in the same formalism, KL-ONE (Brachman 1978), as the system’s knowledge base. The semantic interpreter uses *projection rules* to form an interpretation in a language called JARGON, which is then translated into KL-ONE. Bobrow and Webber are particularly concerned with using this framework to determine the combinatoric relationship between quantifiers in a sentence.

Bobrow and Webber’s approach addresses several of the issues that we do, in particular the relationship between syntax and semantics. The information feedback to the parser is similar to our Semantic Enquiry Desk, though in our system, because the parser is deterministic, semantic feedback cannot be conflated with syntactic success or failure. Both approaches rely on the fact that the objects manipulated are objects of a knowledge representation that permits appropriate judgments to be made, though in rather a different manner.

Hendler and Phillips (1981; Phillips and Hendler 1982) have implemented a control structure for NLU

¹⁸Gawron *et al* produce all possible trees and their translations for the input sentence, and then throw away any that don’t make sense to the database.

¹⁹Our choice of Paragram was largely pragmatic—it was available—and does not represent any particular commitment to transformational grammars.

based on message passing, with the goal of running syntax and semantics in parallel and providing semantic feedback to the parser. A “moderator” translates between syntactic constructs and semantic representations. However, their approach to interpretation is essentially ad hoc (James Hendler, personal communication), and they do not attempt to put syntactic and semantic rules in strict correspondence, nor type their semantic objects.

None of the work mentioned above addresses issues of lexical ambiguity as ours does, though Bobrow and Webber’s incremental description refinement could possibly be extended to cover it. Also, Gawron *et al* have a process to disambiguate case roles in the logical form after it is complete, which operates in a manner not dissimilar to the case-slot part of Polaroid Words.

8. Conclusion

We have described a new approach to semantic interpretation, one suggested by the semantic formalism of Richard Montague. We believe this work to be a clean and elegant foundation for semantic interpretation, in contrast to previous ad hoc approaches. At the moment, though, the work is only a foundation; the test of a foundation is what can be constructed on top of it. We do not expect the construction to be unproblematic; here are some of the problems we will have to solve.

First, the approach is not just compositional but almost *too* compositional. At present, noun phrases are taken to be invariably and unalterably specific and extensional, that is to imply the existence of the unique entity or set of entities that they specify. In English, this is not always correct. A sentence such as:

Nadia owns a unicorn.

implies that a unicorn exists, but this is not true of:

Nadia talked about a unicorn.

which also has a non-specific reading. Montague’s solution to this problem does not seem easily adaptable

to Absity.²⁰ Similarly, a sentence such as:

The lion is not a beast to be trifled with.

can be a generic statement intended to be true of all lions; Montague did not treat generics.

Second, the approach is heavily dependent upon the expressive power of the underlying frame language. For example, our language, Frail, is yet deficient in its handling of time, and this is clearly reflected in Absity. Further, the approach makes certain claims about the nature of frame representations—that a descriptive adjective in some sense is a slot-filler pair, for example—that might be shown to be untenable.

We will also have to deal with problems in quantification, anaphoric reference, and many other areas. Nevertheless, we believe that this approach to semantic interpretation shows considerable promise.

Acknowledgements

I am grateful to Eugene Charniak, Carole Chaski, Jim Hendler, Polly Jacobson, and Nadia Talent for their comments upon earlier versions of this paper.

References

- BOBROW, Robert J and WEBBER, Bonnie Lynn (1980a). "PSI-KLONE: Parsing and semantic interpretation in the BBN natural language understanding system." *Proceedings of the Third Biennial Conference, Canadian Society for Computational Studies of Intelligence / Société Canadienne pour Études d'Intelligence par Ordinateur*, Victoria, May 1980. 131-142.
- BOBROW, Robert J and WEBBER, Bonnie Lynn (1980b). "Knowledge representation for syntactic/semantic processing." *Proceedings of the First Annual National Conference on Artificial Intelligence*, Stanford, August 1980. 316-323.
- BRACHMAN, Ronald J (1978). "A structural paradigm for representing knowledge." Report 3605, Bolt, Beranek and Newman, Cambridge, MA 02138. May 1978.
- CHARNIAK, Eugene (1981). "A common representation for problem-solving and language-comprehension information." *Artificial Intelligence*, 16(3), July 1981, 225-255.
- CHARNIAK, Eugene (1983). "A parser with something for everyone." [1] in: King, Margaret (editor). *Parsing natural language*, London: Academic Press, 1983. [2] Technical report CS-70, Department of Computer Science, Brown University, Providence, RI 02912. April 1981.
- CHOMSKY, Avram Noam (1965). *Aspects of the theory of syntax*. Cambridge, MA: The MIT Press, 1965.
- DOWTY, David R; WALL, Robert Eugene and PETERS, Stanley (1981). *Introduction to Montague semantics* (= Synthese language library 11). Dordrecht: D. Reidel, 1981.
- FODOR, Jerry Alan (1978). "Tom Swift and his procedural grandmother." *Cognition*, 6(3), September 1978, 229-247.
- FODOR, Jerry Alan (1979). "In reply to Philip Johnson-Laird." *Cognition*, 7(1), March 1979, 93-95.
- FRIEDMAN, Joyce; MORAN, Douglas Bailey and WARREN, David Scott (1978a). "Explicit finite intensional models for PTQ." [1] *American journal of computational linguistics*, 1978:1, microfiche 74, 3-22. [2] Paper N-3, Computer Studies in Formal Linguistics, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor, MI 48109.
- FRIEDMAN, Joyce; MORAN, Douglas Bailey and WARREN, David Scott (1978b). "An interpretation system for Montague grammar." [1] *American journal of computational linguistics*, 1978:1, microfiche 74, 23-96. [2] Paper N-4, Computer Studies in Formal Linguistics, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor, MI 48109.
- FRIEDMAN, Joyce; MORAN, Douglas Bailey and WARREN, David Scott (1978c). "Evaluating English sentences in a logical model: A process version of Montague grammar." [1] *Proceedings of the 7th International Conference on Computational Linguistics*, Bergen, Norway, August 1978. [2] Paper N-15, Computer Studies in Formal Linguistics, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor, MI 48109. August 1978.
- GALLIN, Daniel (1975). *Intensional and higher-order modal logic with applications to Montague semantics* (= North-Holland Mathematics Series 9). Amsterdam: North-Holland, 1975. [Revised from the author's doctoral dissertation, Department of Mathematics, University of California, Berkeley, September 1972.]
- GAWRON, Jean Mark; KING, Jonathon J; LAMPING, John; LOEBNER, Egon E; PAULSON, E Anne; PULLUM, Geoffrey K; SAG, Ivan A and WASOW, Thomas A (1982). "Processing English with a generalized phrase structure grammar." [1] *Proceedings, 20th Annual Meeting of the Association for Computational Linguistics*, Toronto, June 1982. 74-81. [2] Technical note CSL-82-5, Computer Science Laboratory, Hewlett-Packard, Palo Alto, CA 94304. April 1982.
- GAZDAR, Gerald (1982). "Phrase structure grammar." in: JACOBSON, Pauline Ida and PULLUM, Geoffrey K. *The nature of syntactic representation*. Dordrecht: D. Reidel, 1982.
- GODDEN, Kurt Sterling (1981). *Montague grammar and machine translation between English and Thai*. Doctoral dissertation, Department of Linguistics, University of Kansas, 1981.
- HAMBLIN, C L (1973). "Questions in Montague English." [1] *Foundations of language*, 10(1), May 1973, 41-53. [2] in Partee 1976, 247-259.
- HENDLER, James Alexander and PHILLIPS, Brian (1981). "A flexible control structure for the conceptual analysis of natural language using message-passing." Technical report TR-08-81-03, Computer Science Laboratory, Texas Instruments Incorporated, Dallas, TX 75266, 1981.
- HIRST, Graeme (1983). *A foundation for semantic interpretation, with word and case disambiguation*. Doctoral dissertation, Department of Computer Science, Brown University [forthcoming].
- HIRST, Graeme and CHARNIAK, Eugene (1982). "Word

²⁰He handled such sentences by having two distinct parses, one for each reading; a meaning postulate equates the representations of the two parses where the verb makes it appropriate to do so.

- sense and case slot disambiguation." *Proceedings of the National Conference on Artificial Intelligence*, Pittsburgh, August 1982. 95-98.
- HOBBS, Jerry Robert and ROSENSCHEIN, Stanley Joshua (1977). "Making computational sense of Montague's intensional logic." *Artificial Intelligence*, 9(3), December 1977, 287-306.
- JOHNSON-LAIRD, Philip Nicholas (1978). "What's wrong with Grandma's guide to procedural semantics: A reply to Jerry Fodor." *Cognition*, 6(3), September 1978, 249-261.
- JONES, Mark A and WARREN, David Scott (1982). "Conceptual dependency and Montague grammar: A step toward conciliation." *Proceedings of the National Conference on Artificial Intelligence*, Pittsburgh, August 1982. 79-83.
- LEHNERT, Wendy Grace and RINGLE, Martin H (1982). *Strategies for natural language processing*. Hillsdale, NJ: Lawrence Erlbaum Associates, 1982.
- MARCUS, Mitchell P (1980). *A theory of syntactic recognition for natural language*. Cambridge, MA: The MIT Press, 1980.
- MONTAGUE, Richard (1973). "The proper treatment of quantification in ordinary English." [1] in: HINTIKKA, Kaarlo Jaakko Juhani; MORAVCSIK, Julius Matthew Emil and SUPPES, Patrick Colonel (editors). *Approaches to natural language: Proceedings of the 1970 Stanford workshop on grammar and semantics*. Dordrecht: D. Reidel, 1973. 221-242. [2] in: THOMASON, Richmond Hunt (editor). *Formal philosophy: Selected papers of Richard Montague*. New Haven: Yale University Press, 1974. 247-270.
- MOORE, Robert C (1981). "Problems in logical form." *Proceedings, 19th Annual Meeting of the Association for Computational Linguistics*, Stanford, July 1981. 117-124.
- PARTEE, Barbara Hall (1973). "Some transformational extensions of Montague grammar." [1] *Journal of Philosophical Logic*, 2, 1973, 509-534. [2] in Partee 1976, 51-76.
- PARTEE, Barbara Hall (1975). "Montague grammar and transformational grammar." *Linguistic Inquiry*, 6(2), Spring 1975, 203-300.
- PARTEE, Barbara Hall (editor) (1976). *Montague grammar*. New York: Academic Press, 1976.
- PHILLIPS, Brian and HENDLER, James Alexander (1982). "A message-passing control structure for text understanding." in: HORECKY, Ján (editor). *COLING 82: Proceedings of the Ninth International Conference on Computational Linguistics, Prague, July 5-10, 1982* (= North-Holland Linguistic Series 47). Amsterdam: North-Holland, 1982. 307-312.
- ROSENSCHEIN, Stanley Joshua and SHIEBER, Stuart M (1982). "Translating English into logical form." *Proceedings, 20th Annual Meeting of the Association for Computational Linguistics*, Toronto, June 1982. 1-8.
- SCHANK, Roger Carl (editor) (1975). *Conceptual information processing* (= Fundamental studies in computer science 3). Amsterdam: North-Holland, 1975.
- SCHANK, Roger Carl (1982). "Reminding and memory organization: An introduction to MOPs." [1] in Lehnert and Ringle 1982, 455-494. [2] Research Report 170, Department of Computer Science, Yale University, New Haven, CT 06520. December 1979.
- SCHUBERT, Lenhart K and PELLETIER, Francis Jeffrey (1982). "From English to logic: Context-free computation of 'conventional' logical translation." *American Journal of Computational Linguistics*, 8(1), January-March 1982, 26-44.
- SMITH, Brian Cantwell (1979). "Intensionality in computational contexts." Unpublished MS, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139. December 1979.
- TARNAWSKY, George Orest (1982). *Knowledge semantics*. Doctoral dissertation, Department of Linguistics, New York University, 1982.
- WILKS, Yorick Alexander (1982). "Some thoughts on procedural semantics." [1] in Lehnert and Ringle 1982, 494-516. [2] Technical report CSCM-1, Cognitive Studies Centre, University of Essex, Wivenhoe Park, Colchester. November 1980.
- WONG, Douglas (1981a). "Language comprehension in a problem solver." *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, Vancouver, August 1981. 7-12.
- WONG, Douglas (1981b). *On the unification of language comprehension with problem solving*. Doctoral dissertation [available as technical report CS-78], Department of Computer Science, Brown University, 1981.
- WOODS, William Aaron Jr (1967). *Semantics for a question-answering system*. [1] Doctoral dissertation, Harvard University, August 1967. [2] reprinted as a volume in the series Outstanding dissertations in the Computer Sciences, New York: Garland Publishing, 1979.
- WOODS, William Aaron Jr (1968). "Procedural semantics for a question-answering machine." *AFIPS conference proceedings*, 33 (Fall Joint Computer Conference), 1968, 457-471.

