6 Natural language, knowledge representation, and logical form

JAMES F. ALLEN

6.1 Introduction

If current natural language understanding systems reason about the world at all, they generally maintain a strict division between the parsing processes and the representation that supports general reasoning about the world. The parsing processes, which include syntactic analysis, some semantic interpretation, and possibly some discourse processing, I will call *structural* processing, because these processes are primarily concerned with analyzing and determining the linguistic structure of individual sentences. The part of the system that involves representing and reasoning about the world or domain of discourse I will call the *knowledge representation*. The goal of this chapter is to examine why these two forms of processing are separated, to determine the current advantages and limitations of this approach, and to look to the future to attempt to identify the inherent limitations of the approach. I will point out some fundamental problems with the models as they are defined today and suggest some important directions of research in natural language and knowledge representation. In particular, I will argue that one of the crucial issues facing future natural language systems is the development of knowledge representation formalisms that can effectively handle ambiguity.

It has been well recognized since the early days of the field that representing and reasoning about the world are crucial to the natural language understanding task. Before we examine the main issue of the chapter in detail, let us consider some of the issues that have long been identified as demonstrating this idea. Knowledge about the world can be seen to be necessary in almost every aspect of the understanding task. Here we consider lexical ambiguity, determining the appropriate syntactic and semantic structures, identifying the referent of noun phrases, and identifying the intended speech act.

This chapter was greatly improved based on feedback and discussion with Andy Haas, Becky Passonneau, Jeffry Pelletier, and Graeme Hirst, and on feedback from talks on early versions of the chapter given at Rochester, BBN Labs, and AT&T Bell Labs. The work was supported in part by ONR/DARPA research contract no. N00014-82-K-0193 and NSF grant IRI-9003841.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

6.1.1 Lexical ambiguity

It is well known that most words in natural language are ambiguous, and this introduces considerable complexity into the understanding task. Some forms of potential ambiguity can be resolved by structural methods: the verb *operate,* for instance, occurs in a transitive form (*He operated the printing press for years*) and an intransitive form (e.g., *The doctor operated on my foot*), and the appropriate word sense can be determined by the syntactic structure of the sentence in which it is used. Other word senses can be identified by exploiting fixed semantic co-occurrence restrictions (i.e., selectional restrictions, as described by Katz and Fodor, 1963). The word *pilot,* for instance, can be a small flame used to start a furnace, a preliminary study, or a person who flies airplanes. But in the sentence *The pilot ate his lunch,* the word appears unambiguously, used in its person sense. This is simply because the verb *eat* requires an animate subject. Selectional restrictions encode such structural restrictions in terms of what are legal combinations of sub-constituents to form new constituents. It is easy to see, however, that such techniques cannot solve all ambiguity resolution problems. Adapting an example from Bar-Hillel (1971), consider the sense of the word *pen* in *I put the pigs in the pen* versus *I put the ink cartridge in the pen.* We must have general knowledge about writing instruments (*pens*) – that they typically contain ink, and that they are not large enough to hold pigs – and about animal compounds (*pens*) – that typically there would be no reason to put ink in one, and that they typically are used to contain animals such as pigs. As we will see in all these motivating examples, the general problem, in this case word-sense ambiguity, cannot be solved without using both structural constraints and general reasoning.

6.1.2 Structural ambiguity

Similar issues arise in dealing with syntactic-structure ambiguity, as in prepositional phrase attachment. There seems to be structural influences that prefer certain attachment assignments. One of the most studied is the preference called minimal attachment, which suggests that prepositional phrases are attached so as to minimize the depth of the final syntactic tree. Therefore, using an example from Schubert, the sentence *I saw the bird with the binoculars* has a preferred reading where binoculars were used to see the bird, rather than being related to the bird in some way. Another preference that has been extensively studied is called *right association,* in which the preferred attachment is to the right-most constituent. Right association would predict the incorrect reading for the above sentence but be correct for the sentence *I saw the bird with the yellow feathers.* Even if we choose a single strategy, or introduce some way to resolve conflicts between these preferences, it still remains that some sentences clearly violate these constraints and the appropriate reading can only be identified by the knowl-

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

edge representation. So the knowledge representation (KR) seems to have the final say. On the other hand, we know that certain structures seem to yield such strong preferences that a comprehensible parse cannot easily be found even though it exists. These are the so-called "garden path" sentences such as *The horse raced past the barn fell* (Bever, 1970). So the structure of the sentence can have great influence on the interpretation, but so can general world knowledge and reasoning. Thus, a reasonable solution must allow the structural constraints and general reasoning about the world to be used together to identify the appropriate reading.

6.1.3 Reference

Another area concerns determining the referent of noun phrases. Consider pronominal reference as an example. Although there may be strong structural suggestions for determining the referent of a pronoun, structure does not completely determine the issue. Hobbs (1978) and Brennan, Friedman, and Pollard (1987), for instance, develop algorithms based on syntactic structure and simple type restrictions that can correctly identify up to 90% of the antecedents in news articles. But examples that cannot be resolved on structural grounds and require general reasoning are easily found. Consider a variant of an example from Winograd: *I dropped the glass on the table and it broke.* The pronoun *it* appears to refer to the glass, just as it does in *I dropped the table on the glass and it broke.* Determining the correct referent in both of these sentences can only be a result of general knowledge about tables and glasses and which are more likely to break in certain situations. On the other hand, general reasoning cannot account for all cases; otherwise the discourse segment (adapted from Wilks, 1975) *I put the wine on the table. It was brown and round* would not seem anomalous. In this case, as Hirst (1981) pointed out, there is a perfectly good referent for the pronoun, namely the table, but the structure of the sentence seems to prohibit this, leaving us with the anomalous situation of having brown, round wine! Grosz et al. (1983) have explored various structural constraints on pronominal reference that may be able to explain such phenomena. But again, this structural model must interact with general reasoning in order to determine the appropriate interpretation.

6.1.4 Speech acts

Another aspect of language that is highly dependent on general reasoning is recognition of the intended speech act. The sentence *Do you know the time?* for instance, can be intended in different situations as a yes–no question, as a request for the time, as an offer to tell someone the time, or as a reminder that it is late. This seems like a situation in which structural processing may be of little value, and general reasoning dominates. In fact, the model of Perrault and Allen (1980),

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

and most of the speech act work undertaken since, has made this assumption explicitly. The intended speech act is determined solely from the propositional content of the sentence, together with minimal structural information, specifically the syntactic mood of the sentence. Perrault and Allen show how a wide range of interpretations can be derived simply from reasoning about the speaker's beliefs, actions, and intentions in the situation. But there are strong structural indicators of the intended speech act as well. For instance, *Can you pass the salt?* is much more likely a request to pass the salt than *Tell me if you are able to pass the salt*. As another example, any sentence with the adverbial *please* must be taken as some form of request. So speech act interpretation is yet one more phenomenon that depends on both structural constraints and general reasoning.

Given these examples, it has been known for years that a central problem in natural language understanding is how linguistic structure and general reasoning interact in order to produce an understanding of the sentence. But progress on this problem has been slight and the usual approach taken is a combination of avoidance and enumeration. First, the test application is carefully constrained to avoid as many ambiguity problems as possible. For the remaining ambiguity, the structural processing enumerates a set of unambiguous interpretations for the knowledge representation to select from. But even after constraining ambiguity extensively, the enumerate-and-filter model is hard to manage in practice. As a result, current systems depend on being able to heuristically resolve the remaining ambiguity during the parsing process, so that a single unambiguous representation of the sentence is produced as the starting point of general reasoning.

This chapter will examine these issues in detail and conclude that the current methods used will not be extendable to a general solution to the problem. Progress on handling ambiguity will occur only after we re-consider the architecture of natural language systems and make a considerable change in emphasis in our research in knowledge representation.

Before I continue with the main thrust of the chapter, I want to eliminate the following possible source of confusion. Could we not view the *entire* language understanding process as reasoning in some knowledge representation? After all, logic-grammar systems have been used for syntactic and semantic processing for a long time now (e.g., Colmerauer, 1978; Pereira and Warren, 1985). These systems can be viewed as theorem provers in a Horn-clause logic, as in the formal development of PROLOG. So is there any issue here? If we all used such systems, then could there be a uniform representation for parsing and general world reasoning?

At a superficial level, the answer to this may be yes. But that only serves further to hide the problems we want to examine, rather than eliminate them. There are significant differences between a parser built in a Horn-clause formalism, and a knowledge representation for general reasoning about the world. In fact, they are two completely different levels of representation, even if they do share certain syntactic similarities to each other. On the one hand, the terms in a

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

logic grammar denote abstract structures in some theory of syntax. The terms in a general reasoning system, on the other hand, denote objects in the world. Even if we allowed the domain of syntactic objects into the domain of our general representation, we would still have the problem of defining the form of these structures (i.e., traditional syntactic theory) and, more important, defining how these syntactic structures relate to the world they describe. This is the same problem with which we started, so collapsing the two representations has not solved anything.

More important, the types of inference allowed at each level of representation are significantly different. In particular, if we view a parser's operation as an inference process, then the inference rules allowed are highly constrained: they all take certain previous "conclusions" (i.e., subconstituents) and combine them syntactically to produce a larger "conclusion" (i.e., the new constituent defined by the rule). Furthermore, the nature of the underlying syntactic theory constrains the types of composition that are allowed. These are the crucial properties that allow us to build efficient parsers. For instance, the compositional nature of this "inference process" allows us to use techniques such as well-formed substring tables and top-down filtering in order to optimize parsing algorithms.

General reasoning, on the other hand, appears to be unconstrained: there is provably no effective algorithm for determining all possible inferences and there certainly is no convenient structural relationship between a conclusion and its premises. Because of this, there is great concern in the knowledge representation community about the trade-off between the expressive power of the formalism and the computational complexity of the reasoning processes (e.g., Levesque and Brachman, 1984). Even highly restricted forms of reasoning, such as determining whether a set of temporal interval constraints is consistent, can be shown to be NP-hard (Vilain and Kautz, 1986). There is certainly no sort of compositionality constraint on general inference that allows us to guarantee efficient algorithms, although a few subareas (e.g., type subsumption in a tree-structured type hierarchy) can be analyzed in this fashion.

Therefore, it does not help to have a uniform representational language that can encode the different levels of representation needed for language understanding. In fact, although it may be a reasonable implementation technique, for the purposes of this chapter it is much more helpful to assume different representation languages for structural analysis and general reasoning, and thus underline this chapter's issues of concern.

6.2 Two views of knowledge representation

Not only do most systems use separate representations for parsing and knowledge representation, they also often have the parser produce an initial meaning representation, or *logical form,* in a different language from the final knowledge representation. The reasons for this reflect the very different requirements on a

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

representation required by researchers in the natural language community and those in the knowledge representation community. The two main representational concerns in the natural language community are expressive power and the handling of ambiguity. Both of these issues push the formalisms used in natural language systems away from those developed in the knowledge representation community. This section considers these issues in more detail.

In natural language work, the knowledge representation must be expressive enough to conveniently capture the types of information conveyed in language, and it is clear that a representation with the expressive power of the first-order predicate calculus is not sufficient. For instance, many verbs in natural language are more closely related to modal operators than standard first-order predicates – beliefs, wants, possibility, and so on. In addition, there is a need to have predicate-modifying operators. For example, if the sentence *The box is red* has some representation of the form RED(B1), then how would one represent the meaning of the sentence *The box is very red*? It could be that there is another predicate VERY-RED, say, that captures the meaning of this sentence, but this seems to miss some important generalizations. We would, for instance, need an extra predicate of the form VERY-P for every predicate P that can be so modified. A much more satisfying representation would involve a predicate operator VERY, producing a meaning representation of the form VERY(RED)(B1).

In addition, there is significant interest currently in defining compositional semantic interpretation algorithms. In these schemes, the semantic representation of one constituent is a function of the semantic interpretation of its sub-constituents. Compositionality is a highly desirable feature in semantic interpretation, as it allows a close correspondence between semantic interpretation and syntactic parsing, and because of this produces a representation that is cleanly defined and generally more easily extended than the non-compositional approaches. But taking this approach makes requirements on the representational language. Returning to the example above, the meaning of the phrase *very red* should be a simple function of the meaning of the sub-constituent *red.* If the representation contains predicate operators, this is trivial: the meaning of *red* might be the predicate RED, and the meaning of *very red* could be the predicate VERY(RED). Consider the alternative. If RED and VERY-RED are distinct predicates, even though they are inferentially related (i.e., very red things are red), they have no systematic connection between each other. Thus the meaning of the phrase *very red* would have no systematic connection to the meaning of the phrase *red,* making a compositional analysis impossible. Compositionality also requires further extensions to the representation. For example, the standard technique for combining sub-constituent meanings into the constituent meaning requires the use of the lambda-calculus, as well as an ability to distinguish extensional and intensional meanings (e.g., Scha and Stallard, 1988).

Another very important representational issue that is often ignored is nominalization. In natural language almost anything can be nominalized. For example, although *red* above was represented as a predicate, we can also talk about it as an object and assert properties of it, as in *Red is my favorite color.* As above, it could be that we have a predicate RED, and an object for the color red, but this misses the significant generality of this phenomenon. A much better representation would involve the introduction of a nominalization operator that can turn a predicate into an object. Moreover, it appears that arbitrarily complex actions and situations can be referred to in natural language. To handle these phenomena adequately, we need to introduce some situation-producing operator as well (e.g., see Schubert and Hwang, 1989).

So natural language researchers require a rich, highly expressive formal representation language in order to capture adequately the meaning of natural sentences. The issue of the complexity of complete reasoning within the representation does not appear as a central issue.

The other central concern in the natural language (NL) community is the handling of ambiguity. Because enumeration does not seem to be a viable strategy, some method is needed of concisely encoding ambiguity until it reaches the stage when it can be resolved. There are many techniques that can be used, and they are often found in various logical forms. Scha and Stallard (1988), for example, introduce a representation, EFL, that encodes the structural semantics of a sentence. EFL contains a term for every word in the lexicon, which in a sense is an ambiguous encoding of all the possible senses for that word. Allen (1987) uses a logical form representation that allows an enumeration of possible values anywhere in the language where a term may appear as a concise encoding of certain simple forms of disjunction. Many researchers (e.g., Woods, 1975; Schubert and Pelletier, 1982; McCord, 1986; Allen, 1987; Hobbs and Shieber, 1987; Alshawi, 1990) use a representation in which quantifier scoping may be undetermined in the logical form. Many of these representations keep the quantifier directly with the rest of the interpretation of the noun phrase. Thus, rather than the sentence *Every boy loves a dog* being ambiguous between the two meanings, say

$$\forall\, b\ .\ BOY(b) \supset (\exists\, d\ .\ DOG(d) \wedge LOVES(b,d))$$
$$\exists\, d\ .\ DOG(d) \wedge (\forall\, b\ .\ BOY(b) \supset LOVES(b,d))$$

the sentence has a single, but ambiguous, representation such as this:

$$LOVES(<\forall\, b\ BOY(b)>,<\exists\, d\ DOG(d)>).$$

Furthermore, the argument structure of this representation corresponds reasonably closely to the syntactic structure of the sentence itself, making a compositional analysis much simpler. Schubert and Pelletier (1982) and McCord (1986) extend this technique further to include other forms of scoping ambiguity, including the negation operator and various modal and adverbial forms.

The knowledge representation community, on the other hand, views model-theoretic semantics and the computational complexity of the inference processes

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

as the primary issues (e.g., Woods, 1975; Levesque and Brachman, 1984). After an initial period of development of informally defined, procedural representation systems, in the mid-1970s the definition of a formal semantics for representational frameworks became a central issue. With the exception of the identification of default inference strategies (e.g., Reiter, 1980; McCarthy, 1980), many representations were found to be expressively equivalent to the first-order predicate calculus once they were analyzed (e.g., Hayes, 1979). Since that time, virtually all new representations have been introduced with a formal semantics, and attention has turned to defining the inferential power of the reasoning component. The simplest and most common method of defining inferential power is to use a complete inference strategy – one that can infer all the logical consequences of the represented knowledge. Unfortunately, representations with the full power of the first-order predicate calculus (FOPC) cannot have a tractable complete inference procedure, and considerable effort has been made recently to deal with this problem.

The general response to this difficulty is to limit the expressive power of the formalism so that the inference procedure becomes tractable. For instance, Levesque and Brachman (1984) and Patel-Schneider (1986) both suggest representations that allow a tractable complete inference procedure. The KRYPTON system (Brachman et al., 1983) isolates a particular form of representation, the terminological component, that also has provably tractable inference procedures. Unfortunately, these representations lack the expressive power required for natural language semantic processing, which as we discussed above, really requires representations more expressive than FOPC. A possible solution to this dilemma is to retain an expressive language, but define a tractable limited-inference procedure on this representation. Frisch (1987) explores such an approach for a language with the expressive power of FOPC.

Another approach to this problem is the development of hybrid representation systems (e.g., KRYPTON [Brachman et al., 1983]). These systems consist of several specialized reasoning systems, each one with well-defined inferential properties, that are combined together to form a representation more expressive than any of its parts. Of course, there is no free lunch, and the overall representational system either has an intractable inference procedure or is incomplete. It is very easy to combine two tractable systems to form an intractable combined system.

Thus the natural language community and knowledge representation community have very different goals: the NL researchers are interested in expressiveness and handling ambiguity, whereas the KR researchers are interested in devising representations with limited tractable inference procedures. This is unfortunate for both camps: only a few natural language researchers are developing representational systems that involve significant inferential power or can represent complex knowledge about the world, whereas the knowledge representation researchers are developing representations that do not address the issues of concern

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

to one of the largest groups of potential users. Given this situation, it is interesting to examine how current natural language systems deal with this dilemma, and this is the subject of the next section.

6.3 Language and knowledge in existing systems

Surprisingly, many current NL systems do not have any general reasoning component at all, so the issues of concern in this paper are avoided altogether. The typical database query systems such as JANUS (Weischedel, 1989) or TEAM (Grosz et al., 1987), and story understanders such as SCISOR (Jacobs and Rau, 1990) do not use any general reasoning. The processing is based entirely on structural knowledge such as syntax, selectional restrictions, case grammar, and static knowledge such as frames. Because of these constraints, such systems are only feasible to highly specialized, limited-application domains where ambiguity resolution can be reduced to structure-based heuristics.

Although systems that do employ a knowledge representation differ radically from one another, they all draw a sharp division between the parsing (i.e., structural) processes and the general inference processes. I will discuss several specific examples in this section to justify this claim, but will start with a discussion of a hypothetical "generic" system as shown in Figure 6.1. This system has a syntactic grammar and a structural semantic interpretation scheme (for example, based on selectional restrictions and case grammar), and produces a representation that we will call the *logical form*. This logical form is then translated into the knowledge representation language using a combination of simple expression-to-expression mapping rules, some structural discourse models, and some highly constrained inference processes such as enforcing-type constraints. This representation then may be used to initiate inference processes such as plan recognition, script/frame matching, question-answering, or other techniques.

Depending on the actual system, the logical form might already be a valid expression in the knowledge representation, and no translation phase is required, or it might be a separate formalism serving as a bridge between the more expressive linguistic representation and a less expressive knowledge representation that supports inference. Systems also differ on whether syntax plays a dominant role in the parsing process, or whether the parser is driven more by a structural semantics (i.e., selectional restrictions and case grammar). Irrespective of all these variations, every system operates in two major phases – the structural processing is done, a logical form produced (and translated into the knowledge representation if required), and then inference is performed on the literal meaning. Of course, the amount of general reasoning required varies significantly from system to system depending on the task that the system has to perform.

Consider JANUS (Weischedel, 1989), a system that has been aimed primarily at database query applications. The parsing and semantic interpretation processes

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

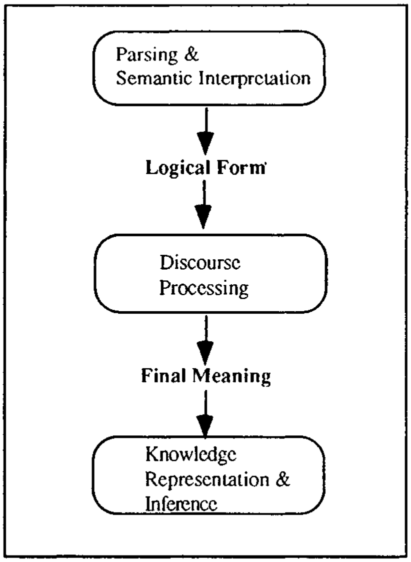

Figure 6.1. A generic natural language understanding system.

derive a formula in an initial representation called EFL, an expressive formalism that concisely encodes word sense ambiguity, and includes many of the features needed to support a compositional Montague-style semantic interpretation. This expression is then translated into an intensional logic meaning representation language, MRL, in which all ambiguity is resolved. This representation is translated into the final representation, be it a limited inference system or a database query language, which then allows for inference and/or retrieval for question answering. No general inference processing by the knowledge representation is involved at any stage of the structural analysis. The semantic interpretation is a structural process closely tied to the parsing process, and the mapping from EFL to MRL uses solely such techniques as static type restrictions to disambiguate the terms. The mapping to the final representation is based entirely on the structural properties of MRL.

The representation in SRI's Core Language Engine (Alshawi, 1990) uses a similar sequence of representations in its analysis. The parser produces a *quasi-logical form* (*QLF*), which is neutral with respect to quantifier scope and certain referential ambiguities. The QLF is then transformed into a *resolved quasi-logical form* (*RQLF*) in which all scoping ambiguity is removed. This is done using structural properties of the sentence and does not involve inference. The final logical form is produced from the RQLF by resolving all the referents for the definite noun phrases. Although this may involve queries to a knowledge repre-

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

sentation to obtain possible referents, it does not involve evaluating the plausibility of different readings for the overall sentence. Furthermore, this representation does not encode lexical ambiguity except by enumerating the lexically unambiguous QLFs, RQLFs, and final logical forms.

The speech act recognition system of Allen and Perrault (1980) produces a logical form solely from syntactic and structural semantic processing. This analysis is then transformed into a literal speech act representation simply based on the syntactic mood of the sentence. Declarative mood sentences become *inform* acts with the logical form as the propositional content, imperative sentences become *request* acts with the logical form as the action requested, and interrogative sentences become *question* acts with the logical form as the propositional content of the question. In this particular system, the logical form is the same as the final knowledge representation. As a result, the complexity of sentences representable is restricted by the limitations of the representation used for plan reasoning. To handle reference, the representation allows definite descriptions that can later be resolved during plan reasoning. Other than this, however, all lexical and structural ambiguity is handled solely by enumeration. Thus it would be difficult to extend directly this system to handle complex sentences with linguistically complex semantic phenomena and extensive ambiguity.

Narrative understanding systems such as SAM (Cullingford, 1981) and PAM (Wilensky, 1983) are organized similarly. Parsing is completely separate from the general reasoning mechanism. It produces a logical form directly in the final knowledge representation that serves as the starting point for inference. Any ambiguity has to be resolved during parsing, and thus could not rely on general inference, although some techniques were developed to allow some influence on parsing by the representation. These techniques, for instance, allow the current active script to identify appropriate word senses at the time the word is initially input.

BORIS (Dyer, 1983) presents a much richer model of interaction between the parser and the reasoning component. In fact, BORIS performs all parsing and reasoning simultaneously. As each word is read, various demons are fired that enforce structural constraints (e.g., thematic role analysis using type restrictions), or perform memory search (for reference), or are designed for disambiguation of particular words. Although this system demonstrates integration, it also shows what the limits are for such an organization. In particular, ambiguity for the most part must be ignored as a research issue – only lexical ambiguity that can be resolved through structural constraints or techniques of matching into memory (into scripts, or lists of objects mentioned recently, for instance) can be handled. There is no capability of retaining ambiguity after the sentence is initially parsed, or of allowing general inference to select the most appropriate sense in context. Furthermore, the syntactic theory is very primitive, and linguistically complex sentences with significant structural ambiguity simply cannot be handled.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

Small and Reiger (1982) define a decision tree for each word that is used to identify the appropriate sense in context, but again the tests are on structural properties of the words rather than general inferential techniques.

Hirst (1987) has the most extensive examination of ambiguity problems in the computational literature. His system uses selectional restrictions, verb case role restrictions, semantic associations (cf. Quillian, 1967; Hayes, 1977), local referent finding techniques to handle lexical ambiguity and some structural ambiguity such as PP attachment. These techniques use the structure of the KR for finding semantic relationships between word senses. The system also allows limited access to the general reasoning component of the KR to resolve ambiguity. In particular, the SED mechanism allows queries to the knowledge representation to help in certain pre-determined disambiguation questions. These queries, however, must eventually result in an unambiguous representation before the processing is completed. Fawcett and Hirst (1986) generalize this approach to allow unambiguous definite descriptions in the final knowledge representation.

The problem every one of these systems is avoiding is that in order to invoke inference, one must start with some proposition to either add or query the knowledge representation system. Although this can be done in certain pre-anticipated cases, say to find all objects in a database that could match a certain definite description, the lexical ambiguities in the descriptions themselves must be resolved before such queries can be made, because the KR does not support ambiguous expressions. So inference based on the meaning of the entire sentence cannot be used to resolve the ambiguities except by enumeration. As a result, existing systems are restricted to using the static structure of the KR (i.e., its type hierarchy and role restrictions, etc.) to find connections between word senses, or to check if a certain predicate (say derived from a PP) can apply to a certain term (say derived from the constituent to be modified). These techniques are very useful, but they do not bring us any closer to a solution to the real problem – how general reasoning in context can be used to resolve ambiguity.

6.4 Must we explicitly represent ambiguity?

As seen above, current systems are organized on the assumption that almost all ambiguity can be resolved before general reasoning is needed, and that the remaining ambiguity, if any, could be resolved by enumeration. In this section, I argue that this is not a viable long-term strategy. In particular, enumeration is much worse a strategy than one might think. In order to use general reasoning to disambiguate sentences, the choices must be represented in the final knowledge representation so that inference is defined on them. Furthermore, in some cases ambiguity may remain over several sentences in a discourse, or never be resolved. In other words, the ambiguity must be representable in the representation that supports inference.

Before we examine the issues in detail, we must distinguish between ambigu-

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

ity and lack of specificity. Both are present in language, and different techniques must be used to handle each. Terms are unspecific to the extent that there are other terms that can be more specific than them. An example is the word *horse,* which is unspecific in the sense that there are other terms that are more specific such as *mare, filly, colt,* etc. Terms are ambiguous when they may denote two or more different things. Thus *pilot* is ambiguous because it can refer to a flame, a study, or a person. Various intuitive tests have been developed in the linguistics literature to distinguish these two (e.g., McCawley, 1981). For example, we can say *Jack bought a horse, and Jill did too,* even in the case where Jack bought a mare and Jill bought a colt. On the other hand, we cannot say (except possibly as a joke) *Jack saw a pilot, and Jill did too* where Jack saw a person flying a plane and Jill saw a flame on a furnace. In this chapter, we are solely concerned with representing ambiguity.

If the necessary techniques for resolving a specific ambiguity could be anticipated in advance, and could be performed reasonably efficiently, then the procedures could be incorporated into the structural processing without requiring the general reasoning abilities of the KR. For example, the major form of disambiguation in current systems depends on the use of co-occurrence restrictions first formulated as selectional restrictions by Katz and Fodor (1963). In the final representation, these restrictions correspond to type restrictions on expressions. Current systems either use redundant representations, encoding such knowledge twice – once in the parser (as selectional restrictions), and once in the KR (as type constraints), or they use the KR type restrictions directly to perform the disambiguation. In the latter case, of course, the words need to have been translated into possible KR expressions before these constraints can be tested. In the former case, the translation into the KR can be delayed because of the redundant encoding of the information as selectional restrictions. If such techniques were sufficient to perform all potential disambiguation, then all ambiguity could be resolved procedurally before the final KR meaning was produced.

Unfortunately, as we have already seen, selectional restrictions are not sufficient. On one hand, they encode only semantically possible interpretations and give no indication of what readings are more likely than others. The sentence *The pilot was out* could either describe a situation in which a person is missing, or describe why a furnace was not working. The only way to decide between these interpretations is to see which best fits the context. No technique based solely on the context-independent meanings of the words can choose the appropriate interpretation. Certainly, other techniques such as spreading activation from concepts mentioned in previous utterances may suggest the more likely interpretation, but these predications can be wrong in some cases, so they only play a heuristic role in choosing which interpretations to enumerate first. The final interpretation still ultimately depends on what makes sense in the context, i.e., the decision is made by the knowledge representation. Furthermore, determining this may involve considerable reasoning about causality, and other forms of reasoning such as plan recognition.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

As mentioned before, current systems handle such problems by enumeration. In the above example, two representations are produced, one for each reading. But this technique will certainly not generalize as other forms of ambiguity need to be considered simultaneously. In particular, if a sentence is made up of n phrases, each k ways ambiguous, then there would be k^n different readings of the sentence to consider! Language is sufficiently ambiguous to make this a truly formidable number. For example, consider quantifier scoping. This is well known to be a complex problem, where there may be certain structural constraints and preferences on the scope. The final determination of the full scoping, if it can be done at all, can only be made by general reasoning about what interpretation is likely. Hobbs and Shieber (1987) give an example of a sentence with five quantifiers that, even after using sophisticated structural constraints, has 42 different readings due solely to scope ambiguity! If we combined scoping ambiguity with word sense ambiguity, or operator scoping, the number of interpretations grows exponentially.

One promising technique used in systems with incremental semantic interpretation is to invoke the knowledge representation to disambiguate each subconstituent as it is formed. If this were feasible, then the knowledge representation would only have to consider k interpretations at any one time, and after n calls, would have a single interpretation of the sentence. Although this technique can be used to advantage, the basic problem remains the same. Many decisions cannot be made locally but require the context of the entire sentence. Reconsider the sentence *We put the pigs in the pen.* Calling the knowledge representation to disambiguate the noun phrase, *the pen,* would not select an interpretation as both readings of *the pen* could be appropriate in some sentence. The appropriate sense of *pen* can only be determined at the sentence level, so we are back to the original situation. Quantifier scoping is another example. It must be determined at the sentence level and cannot be done on a sub-constituent by sub-constituent basis.

The other technique that seems promising in avoiding this issue is to use preferences to order the interpretations and to pass only one reading at a time to the knowledge representation, which then accepts the interpretation if it seems sufficiently reasonable. For example, consider PP attachment in a sentence such as *I saw the bird with yellow feathers.* If we assume a preference for right association, the preferred reading will be the one where the bird has the feathers. This would be the interpretation suggested. Because it is reasonable, it would be accepted by the knowledge representation. The other interpretations would never be considered. On the other hand, with the sentence *I saw the bird with the binoculars,* the initial reading would be that the bird was with the binoculars. If this were deemed unreasonable by the knowledge representation and rejected, a second interpretation, that the seeing was done with the binoculars, would be suggested. This is a fine technique as long as the knowledge representation can decide whether to accept or reject an interpretation without seeing the other interpretations to come. But this seems an impossible task. Even in the above example, the phrase *the bird with the binoculars* could have a reasonable reading

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

in certain contexts: maybe the bird has stolen the binoculars and is flying away with them in its beak. The point here is not that this is likely, but that it is possible, and so the interpretation cannot be eliminated on the grounds of violating some constraint. Rather, it is a matter of which reading is more likely. But if the knowledge representation receives one reading at a time, how can it decide whether ones to follow would be more likely? Schubert (1986) argues that no such serial processing strategy can produce a satisfactory solution to this problem, where he defines a serial strategy to be one that uses preferences to suggest preferred readings in some order, and the first one that is semantically and pragmatically acceptable is chosen.

But what is the alternative to such approaches? It seems that the only one is to encode ambiguity within the knowledge representation language. Now this might seem in one sense simply to be delaying the problem and thus not helping. But this is not necessarily so. What it allows is the ability to do inference on ambiguous sentences, which then may allow us to infer something that eliminates some of the potential readings. Furthermore, the ambiguity would have to be encoded in a way that is not combinatorially explosive, so that options reveal themselves as a result of inference, rather than inference being used to select from a long list of options. For example, one encoding of ambiguity that would not help would be a disjunction of the enumerated possible unambiguous readings!

6.5 Can a logical form save us?

In Section 6.3, we saw that existing systems either artificially limit the amount of ambiguity and sheer complexity of the input, and parse directly into the knowledge representation, or they use an intermediate logical form level to build a bridge between the structural processing and the knowledge representation. The logical form is typically defined to be the context-independent, or literal, meaning of the sentence (e.g., Allen, 1987; Scha, 1983; Moore, 1981). In practice, it is used in order to resolve the incompatibilities that arise between the needs of parsing and semantic interpretation and the need to support general inference, as described in Section 6.2. In this section I examine whether this technique of introducing an intermediate representation, the logical form, has potential to resolve the difficult issue of handling ambiguity. Unfortunately, I will conclude that it does not and that there is no long-term gain from having a logical form representation as distinct from the knowledge representation, if the logical form is taken as a literal meaning of the sentence.

This is not to say that separate logical forms are not useful in the short-term. The division of the structural and reasoning processes allows progress to proceed in each area without being constrained by the other. For instance, the researcher building a language system can design a richly expressive logical form language (e.g., Scha and Stallard, 1988) without worrying about designing a tractable, complete inference system. Since producing a logical form is a well-defined ending-point for a research project, the fact that we then lose expressive power by the partial translation into the simpler knowledge representation language in an actual implementation may not be of theoretical concern. Knowledge representation researchers, on the other hand, can examine issues of representation without the requirement of satisfying a demanding user, namely the NL system. Although this may have been a reasonable initial research strategy, the dangers of the approach should be apparent. NL people may design representations that could never be used to support inference about the world, and KR people may design representations that few natural language researchers will find useful. Clearly if general reasoning is ever going to be able to assist to disambiguation, some middle ground is essential.

The principal long-term issue is this: will the approach of using a separate logical form allow us to define a system with significant ambiguity resolution between the parser and the general reasoner, but not require us to represent ambiguity in the knowledge representation? I am going to argue that it is not, and that for our purposes the logical form should be in the same language as the knowledge representation. Depending on one's preference, one can view this as saying we need to generalize knowledge representation languages to the full expressive power found in logical forms, or one can view it as saying we need to define inference processes on the logical form representation. Either way, we end up in the same place. The argument depends on examining the main reasons that a logical form might help us, and showing that if we take ambiguity seriously, none of the reasons are valid in the long term. In particular, the logical form representation is used for the following reasons:

1. to encode semantic ambiguity concisely and to support the disambiguation process;
2. to obtain the expressive power needed, which is not available in current knowledge representations;
3. to support compositional semantic interpretation.

In response to these needs, I will argue the following three points below:

1. the KR must support long-term representation of ambiguity anyway;
2. the KR cannot avoid having full expressive power;
3. compositional semantic interpretation need not be constrained by the final KR language.

If I can establish these points, then there is little advantage to having a logical form that is distinct from the knowledge representation. In fact, a separate representation could only serve to complicate the issues. In effect, all the issues that drove researchers to defining a logical form in the first place must eventually be faced in the knowledge representation itself if ambiguity is to be taken seriously. Given this conclusion, it seems that a central problem for natural language research in the next decade should be the development of representations

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

that are considerably more expressive than the current KR systems. In addition, because such representations are assured of having intractable complete inferential properties, considerable work must be done on defining limited inference systems that do not depend on the limited expressive power of the representation.

Having a logical form that supports ambiguity does not eliminate the need for the knowledge representation to handle ambiguity as well. To see why, consider what would happen if the KR did not represent ambiguity. In order to disambiguate the logical form, it must be closely tied to the final representation. For example, to make an appropriate prepositional phrase attachment decision, we will need to consider a representation of the prepositional phrase and the potentially modified expression to decide if the attachment makes sense or is expected in the situation. But to do this will require inference in the KR, and thus the logical form of the prepositional phrase and of the potentially modified expression must be translated into the KR (resolving any ambiguities) before the decision can be made. If we have to go to all this work each time, it is not clear what role the logical form (LF) plays in the process at all. As another example, determining the referent of the pronoun *it* in the sentence *I dropped the glass on the table and it broke* requires general reasoning about the likelihood of a glass breaking a table in contrast to a table breaking a glass. Again the LF is not useful in this situation. Rather it concerns the actual object classes (i.e., tables and glasses) in the KR, and what knowledge is known about such situations (i.e., dropping things). If there is a logical form for *it broke,* then it must be translated into a KR representation with the *it* remaining ambiguous before the question of resolving the pronoun can be considered.

In other cases, ambiguity may not be resolvable at all at the sentence level, and may extend indefinitely through a discourse. In fact, it is easy to find conversations where a certain sentence is ambiguous, and the conversation continues in an effort to define the appropriate reading through a clarification subdialog. In other cases, an ambiguity may be present that is never resolved. In response to this problem, one might suggest a model in which sentences are not encoded into the KR until they are fully disambiguated by further conversation, but this is untenable. Because no inference could ever be made from such sentences until they were disambiguated, this would prevent the reasoning system from finding the connections between sentences that are necessary to disambiguate the sentences in the first place!

As an example, consider the sentence *Jill is visiting us next Friday.* It may be ambiguous as to whether the speaker meant this coming Friday (say tomorrow), or a week from that date. Now it is not hard to construct reasonable scenarios in which this ambiguity remains for an indefinite period of time. We might not know the correct referent until the day arrives, or we might never resolve it. We can still perform inference based on this sentence, however – and, say, infer that Jill likes us, or that we will be seeing her sometime in the next two weeks – but the reference of *next Friday* remains undetermined. Hirst (1990) suggests the

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

example *John bought a vxzfl,* where *vxzfl* is an unknown word. Even so, this sentence could be stored in the knowledge base and inference could be performed. For example, a reasonable system should be able to answer the question *Did he pay for it?*

Finally, it may be the case that a speaker is intentionally ambiguous or vague in a sentence, and that the hearer recognizes this and draws certain conclusions from it (say that the speaker is being evasive and does not want to reveal some implication of the true meaning). If we are ever going to be able to model this sort of interaction, the KR not only needs to be able to represent ambiguity, but it must be able to represent explicitly that sentences are ambiguous, and that such ambiguity may be intended! This is clearly beyond our present capabilities but remains a very interesting long-term research problem.

6.5.1 Expressive power

I have just argued that the logical form representation does not remove the need to represent ambiguity in the KR, or help in mediating between the parsing processes and the KR reasoning. This section argues that the KR language must be as expressive as any logical form anyway, eliminating the second reason that one might wish to retain a logical form representation. This argument rests on the point that every distinction representable in the logical form may make a difference to some inference process, and so needs to be captured in the knowledge representation.

We have already argued that the logical form must represent modal operators. But the need for such operators does not disappear once the meaning is encoded in the knowledge representation. Propositions involving modal operators for belief, want, and possibility, for instance, are crucial in reasoning about the speaker's intentions and in determining appropriate responses in dialog (e.g., Allen and Perrault, 1980).

The argument given earlier for needing predicate operators in the LF to capture significant generalizations in language applies just as well to the KR, where the same information must be represented. The representation of *very,* for instance, cannot be encoded by some absolute value on some intensity scale – rather its interpretation depends on the property it modifies, and the context in which it is used. As before, determining a precise interpretation depends on general reasoning in context. In addition, it may be that a precise interpretation cannot be derived and exact implications of the term remain vague. In either case, the KR needs some representation of the vague reading of *very,* for which the predicate operator representation seems as direct as any could be.

Nominalizations play an important role in reference processing. They also are useful to capture the meanings of common sentences, such as the relation between *The book is red* and *Red is my favorite color.* Although the KR could represent the property and the nominalization independently, it would lose signif-

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

icant generalizations on the types of inferences that can be made to relate such sentences. In addition, it is not clear how the nominalization of complex situations could be handled in a general way without such an operator. For example, consider the discourse fragment *Everybody in the room saw Jack kiss Mary. It embarrassed them greatly.* The referent of *it* is the situation of everybody seeing the kiss, and this situation needs to be explicitly represented in the discourse model so that the anaphora resolution algorithm works. But, in addition, this information needs to be represented in the KR. What is the object that appears in the assertion about John and Mary being embarrassed – it must be the situation again, represented in the knowledge representation. The system explicitly needs such constructs to be able to answer a question such as *What was embarrassing?* with an answer such as *The fact that everyone saw John and Mary kiss.* The different constructions that appear in language reflect strongly on the structure of the underlying reasoning system. Note also that the pronoun *them* in the above sentence is ambiguous: it could be John and Mary, or everybody in the room. This is another example of an ambiguity that may not ever be resolved. Consider that I have just spent a paragraph talking about an ambiguous sentence without the ambiguity ever creating a problem!

It can also be easily shown that other distinctions such as the generic/specific distinction, or the collective/distributive distinction for plural NPs have great effect on the inferences that can be made about the situations described. Thus these must be represented in the final KR language as well. So the KR must eventually be able to represent all the complexities that arise in language and the logical form is not needed.

6.5.2 Compositional semantic interpretation

The final argument for logical form is that even if the knowledge representation were fully expressive and supported ambiguity, a separate logical form might still be needed in order to support compositional semantic interpretation. Although there is no conclusive argument to make here, I find this unlikely.

Before discussing this final argument, however, consider what is meant by compositional semantic interpretation. Compositionality requires that the semantic interpretation can be defined incrementally, where the interpretation of each constituent is a function of the interpretations of its sub-constituents. The strongest form of this is found in Montague-style semantics, where the semantic interpretation of a constituent was derived by applying a function (which was the meaning of the first sub-constituent) to the interpretation of the other sub-constituents. But systems may be compositional and not be of this flavor. Many computational systems, for instance, annotate each syntactic rule with a definition of how the semantic interpretation is to be computed from the interpretation of the sub-constituents (e.g., Schubert and Pelletier, 1982). This allows a much simpler semantic form in most cases, yet retains the full power of the original approach.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

Another form of compositionality is based on unification, where the interpretation of a constituent is defined by a set of unification equations showing how to combine the interpretations of the sub-constituents. As the methods of defining the interpretation become more flexible and powerful, the constraints on the form of the representation become less restrictive.

In addition, all the requirements discussed in the last section involved extensions to the KR that brought it into closer correspondence with the constructs present in language. Each one of these additions makes compositional semantics easier. For example, a KR supporting predicate modifier directly mirrors the structural relationships found with adverbials such as *very*. In other cases, the extensions needed to encode ambiguity concisely simplify matters. For instance, a representation that allows unscoped or partially scoped quantifiers requires a mechanism much more general than the simple structural encoding of quantifier scope found in FOPC. With this more general representation, it may be that the interpretation of noun phrases can be encoded locally in the representation in parallel to the syntactic structure. In the following section I briefly discuss one such technique that allows this.

Although it is not possible to give a definitive answer, it appears that there is much to be gained by assuming that there is no separate logical form, at least in its role as a literal meaning representation, and that we should be exploring ways of extending KRs in ways that support compositional semantics.

The final argument for maintaining a separate logical form even if the KR supports ambiguity is that the logical form might simplify further processing by eliminating various syntactic complexities that do not affect general reasoning. But this does not seem possible, as I would claim that virtually every syntactic distinction may have an influence on the reasoning system. Consider speech act disambiguation as an example. Perrault and Allen (1980) showed that certain indirect speech act readings can be obtained by general inference from an initial literal reading of the sentence. But this approach ignores many of the subtleties that appear in general conversation. For example, in many systems, the logical form of the sentences *Can you lift that rock?, Are you able to lift that rock?,* and *Tell me whether you can lift that rock?* are identical. Thus if the plan reasoner in the KR can infer an indirect reading for one, it would also be able to do it for the other. Clearly, this is not always desirable. Furthermore, there can be adverbial modifiers that do not affect the logical form in that they make no truth-theoretic contribution to the sentence. The adverbial *please,* for instance, does not affect the truth conditions of the sentence, yet it should restrict the KR from deriving interpretations that are not in the general class of requests (the directive class in the taxonomy of Searle [1975]). Potentially, arbitrary syntactic structures might be used conventionally to have some intentional significance. Therefore the logical form would need to be able to encode these subtleties, or the KR needs to be able to operate directly from information in the syntactic structure. Either way the logical form seems redundant: if it encodes the entire syntactic structure then

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

it hasn't simplified the later processing, and if the reasoning system can access the syntactic form directly then the logical form yields no advantage.

So on one hand each part of the logical form must have a close correspondence to some expression in the KR (for disambiguation), and on the other hand it must encode the syntactic structure of the sentence. It is not clear how designing such a representation would give us an advantage over working directly with the syntactic structure and the KR alone. It is possible, however, that the notion of syntactic structure might be usefully generalized to include certain semantic phenomena, which we might then call a logical form. But this representation is not the meaning of the sentence any longer, it is the structure of the sentence. Schubert and Pelletier (1982) suggest a logical form representation that seems closest to this approach, and the technique appears to be very useful.

6.6 Ambiguity and knowledge representation

Given that we have established that the knowledge representation must be able to support reasoning under ambiguity, what techniques seem promising for allowing this extension? In this section, I explore some possibilities.

There are three requirements for a representation of ambiguity that seem most important:

1. The KR must distinguish between ambiguity and disjunction;
2. The KR must support inference over ambiguity; and
3. The KR must support disambiguation techniques.

These three criteria are discussed in more detail below, and then I will examine some techniques that may prove useful in building such a representation.

6.6.1 Ambiguity and disjunction

There is an important distinction between an ambiguous statement and a corresponding statement that is a disjunction of non-ambiguous statements, although they can be easily confused as they are so closely related. In particular, the logical consequences of an ambiguous formula appear to be the same as the logical consequences of its corresponding unambiguous disjunctive formula. For example, let α be an ambiguous term that could refer to one of the objects a_1, a_2, . . . , a_n. In this case, the logical consequences of the formula p_α would appear to be the same as $p_{a1} \vee p_{a2} \vee \ldots \vee p_{an}$. For example, let α be the ambiguous referent of the NP *the block,* which might be ambiguous between a toy block relevant in the current context (say *a1*) or a city block relevant in the context (say *a2*). Now the sentence *The block is long* would be *Long*(α) in the ambiguous representation and *Long*(*a1*) $\vee$ *Long*(*a2*) in the unambiguous representation. But if these two formulas were truly equivalent, then the sentences *The block is long* and *Either the toy block is long or the city block is long* would have identical meanings. But there is a big difference. In the former, the speaker knows which block is being talked about, but the hearer does not, whereas in the latter the speaker does not know which block is long either. Although this may seem to be splitting hairs, this distinction could make a large difference to processes such as plan recognition where the speaker's intent must be characterized. We could capture this distinction by introducing a modality to capture the speaker's intended meaning. If we assume such an operator *I* (which we might take as having a modal S5 structure as a rough start), then the distinction between the two sentences above is the distinction between

$$\exists\beta((\beta=b1) \vee (\beta=b2)) \wedge I(L(\beta))$$

and

$$I(L(b1) \vee L(b2)).$$

This is not to say that a KR must represent ambiguity by using such an operator, but it must be able to make the distinction. Such a distinction could make a significant difference to reasoning systems that perform plan recognition and other reasoning tasks to further interpret the utterance.

6.6.2 Direct inference on ambiguous terms

For the ambiguous representation to be any advantage over a straight enumeration technique, the KR must support inference from ambiguous formulas to produce ambiguous conclusions. For instance, let α be an ambiguous formula with the corresponding disjunctive formula $A1 \vee A2 \vee \ldots \vee An$. If we have an inference rule R that can apply to any disjunct *Ai* and produce a conclusion *Bi*, then R should also apply directly to α to produce an ambiguous conclusion β that corresponds to $B1 \vee B2 \vee \ldots \vee Bn$. This property would allow an inference rule to operate on ambiguous sentences to produce an ambiguous conclusion and, in a very real sense, collapse an entire set of inferences corresponding to applying the rule to each unambiguous disjunct. If this can be done successively, we could possibly gain an exponential speedup in processing.

In addition, if the KR supports reasoning over ambiguous assertions, then there is not such a need to resolve completely the ambiguities within a sentence. Rather, some ambiguities might only be resolved "on demand", i.e., if they are needed in order to accomplish some goal of the understander. Ambiguities irrelevant to the goals of the understander might remain unresolved and even unnoticed! Hirst (1990) suggests some ideas along these lines for mixed-depth text understanding.

6.6.3 Disambiguation inference

Finally, the KR must support inferences that serve to disambiguate, or partially disambiguate, formulas. In particular, given an ambiguous formula α and its corresponding disjunctive formula $A1 \vee A2 \vee \ldots \vee An$, if an inference opera-

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

tion R can eliminate some A_i from the disjunct, then the result of applying R to α should be a formula β where the disjunct has been removed. The simplest example involves the resolution rule: Given $A_1 \vee A_2 \vee \ldots \vee A_n$ and $\sim A_i$, the resolution rule concludes $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \vee \ldots A_n$. What we would like is to be able to apply this operation directly to α to yield an ambiguous conclusion β corresponding to $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \vee \ldots \vee A_n$.

Although no current representation systems maintain a distinction between ambiguity and disjunction, some techniques already exist in these systems that are not far from meeting the other criteria for certain classes of ambiguity. I will look at a few of these techniques here.

One technique that can be quite effective for referential ambiguity involves allowing reasoning about equality and inequality. Many implemented knowledge representations impose the restriction that all lexically distinct constants refer to distinct objects, i.e., they support no facility for adding equalities between terms, and hence also avoid the issue of reasoning about distinctness as well. Unfortunately, with such a representation, the NL system must precisely determine the referent of its terms before it is added to the KR. Thus, while the sentence is processed, the NL system must either commit a term, say the noun phrase, *the happy cat,* to refer to an existing constant, say *CAT37,* or the system must create a new constant (say *CAT58*) that is distinct from all other cats in the knowledge representation. A KR that uses equality, however (e.g., RHET, Allen and Miller, 1989; Vilain, 1985), or limited equality reasoning (e.g., Charniak et al., 1985), does not force this problem. A new constant, *CAT58,* may or may not be equal to another constant already in the KR.

A KR with equality can support an approach to ambiguity that meets the inferential properties described above. In particular, certain inferences can be made directly from the representation using the new constant without knowing how it relates to the other cats in the KR. For instance, we could let our ambiguous term α be *CAT58,* and the corresponding disjunction would involve an enumeration over all known cats, e.g., $CAT58 = CAT1 \vee CAT58 = CAT2 \vee \ldots \vee CAT58 = CATn \ldots$ If we later add an equality relation, say $CAT37 = CAT58$, then we would have disambiguated α. But inference can be performed without having to disambiguate the term. A universal statement such as *All cats have four legs* can be applied to α directly without considering the disjunction. General disambiguating inferences are also possible: say we know *Calico(CAT58),* then an equality between α and a non-calico cat would be inconsistent. Furthermore, if we apply the axiom that all calico cats are necessarily female then any inequality between α and a male cat would be inconsistent. Thus, applying this rule to α would implicitly eliminate all male cats from the enumerated disjunction. Although this technique satisfies the two inferential properties described above, it does not explicitly distinguish between the differing intentions underlying an ambiguous term and its enumerated disjunction. If this distinction is important for a given application, then the representation requires extending. Otherwise, it meets the requirements nicely, for one never needs to construct explicitly the enumerated disjunction.

Such a technique can also greatly simplify the semantic interpretation process because it can eliminate large classes of ambiguity problems that need not be resolved before the consequences of the sentence can be explored in context. All that is required of the semantic interpreter is that it encodes whatever constraints are imposed on the interpretation by the structural properties of the sentences. For instance, if we are given the sentence *The old man saw him,* the NL system may create a new constant, say *M44,* to represent the referent of the NP *the old man,* and another, say *M45,* to represent the referent of the pronoun. We would also add $MAN(M44) \wedge OLD(M44)$ and $MALE(M45)$ based on the structure of the sentence. In addition, constraints on the use of a non-reflexive pronoun would require us to add the constraint that *M44* does not equal *M45*. Thus the "ambiguity" and the structural constraint are captured simply and directly in this equality-based representation.

Other sources of ambiguity seem amenable to similar techniques, once the representational details have been worked out. For example, consider the collective/distributive distinction, as in the sentence *Two men lifted the piano* – did they lift it together, or did they both do it separately? The ambiguous form of this sentence would entail that the two men were both involved in piano-lifting acts, and consequences could be drawn from this, even though it is not known if they acted together or if there were two separate acts. Alshawi (1990) describes many phenomena along these lines that could be candidates for extensions to the KR.

Another example of this is quantifier scoping. A suggestion that has appeared many times in the literature is to extend the representation to allow unscoped quantifiers such as

$$LOVES(<\forall\ b\ BOY(b)>,<\exists\ d\ DOG(d)>).$$

The set of models that satisfies such a formula is the union of the models that satisfy each of the unambiguous formulas allowed. In other words, such a formula would correspond to the disjunction

$$(\forall\ b\ .\ BOY(b) \supset (\exists\ d\ .\ DOG(d) \wedge LOVES(b,d))) \vee (\exists\ d\ .\ DOG(d) \wedge (\forall\ b\ .\ BOY(b) \supset LOVES(b,d)))$$

Under reasonable assumptions about the domain, this disjunction is logically equivalent to the formula

$$\forall\ b\ .\ BOY(b) \supset (\exists\ d\ .\ DOG(d) \wedge LOVES(b,d))$$

Thus all conclusions that can be drawn from this formula should be valid conclusions from the ambiguous formula. But this does not mean that the ambiguous expression can be collapsed to this unambiguous form. A sentence later on could still result in this ambiguous interpretation acquiring the "stronger" interpretation, namely

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

$$\exists\, d\ .\ DOG(d) \wedge (\forall\, b\ .\ BOY(b) \supset LOVES(b,d))$$

These are very simple examples, but they show that such techniques for dealing with ambiguous expressions are possible. One other way of enumerating possibilities implicitly is to allow the ambiguous terms into the language as predications and encode the ambiguity using axioms. For example, the ambiguous word *pit* might be represented by introducing three predicates, *PIT*, *FRUIT-PIT*, and *HOLE-IN-GROUND* and the following axioms:

$$\forall\, x\ .\ FRUIT\text{-}PIT(x) \supset PIT(x)$$
$$\forall\, x\ .\ HOLE\text{-}IN\text{-}GROUND(x) \supset PIT(x)$$
$$\forall\, x\ .\ PIT(x) \supset (FRUIT\text{-}PIT(x) \vee HOLE\text{-}IN\text{-}GROUND(x))$$

With this, the word *pit* would initially map into a description involving the predicate *PIT*, and then inference could identify the correct interpretation later if one reading was inconsistent. This could prove to be a powerful technique if systems can be developed that can perform efficient inference over axiom clusters like the above. It is more attractive than enumeration because of its ability to represent more subtle distinctions and interactions, and its ability to leave the ambiguity implicit until some inference process operates on it.

Examining techniques like this is going to be crucial. Not only do such representations allow us to non-combinatorially encode ambiguity, they provide a concise representation to be used as a start for interpretation in context. For instance, some plan recognition technique might be used, starting from this form, that results in selecting an interpretation that best fits the context in terms of what is known about the speaker's beliefs and goals.

Forms involving lexical ambiguity require more complex techniques to be developed. But in certain cases, the ambiguity may be maintained by a generalization of the technique based on equality. For instance, an ambiguous headword in a definite noun phrase might be handled by creating a constant with a disjunctive type. The NP *the report* might refer to a document (say, type *DOC*) and the sound of a gun (say, type *BANG*). We could encode this in the KR by creating a constant *REP89*, say, with the constraint $DOC(REP89) \vee BANG(REP89)$. Again, in integrating the expression into context, or in processing a subsequent sentence, the inappropriate sense might be eliminated, disambiguating the intended meaning. Of course, many knowledge representations do not allow such constraints, and a constant cannot be created without specifying its immediate type. So this is another case where assuming a more general representation can simplify the semantic interpretation. Note that this is an enumeration technique, and therefore it will only be viable if the ambiguity can be successfully localized in the representation and does not interact badly with other forms of ambiguity.

Other forms of lexical ambiguity are harder to handle, and new techniques need to be developed. It is much more difficult to capture verb ambiguity, for instance, because the different senses of the verb might impose different structural interpretations of the sentence. Thus the verb sense ambiguity is not easily localized. It is hoped that situations like this are rare, as the structure of the sentence itself serves as a strong selector of the verb sense. The remaining senses may all involve the same structural analysis of the sentence. It is very important to distinguish the different types of ambiguity that may occur. Certain senses appear to be semantically unrelated to each other, as though it is just chance that the same word can refer to both. In many other cases, the senses are semantically related to each other, and represent variations on some common semantic theme. In these latter cases, a very promising technique is to encode the ambiguity by a generality. In particular, if the representation supports an abstraction hierarchy, and the different semantically related senses all share a reasonable common parent, then we could use the parent type as a generality encoding of the sentence meaning. As before, we may be able to draw conclusions from this representation at the abstract level, and later inference processes might eventually identify the particular interpretation intended. But even if it is never disambiguated, it is still a fine representation of the sentence. Of course, we may lose information. Say we have an abstract type *A*, with three subtypes, *A1*, *A2*, and *A3*, and a verb that is ambiguous between senses *A2* and *A3*. If we use *A* as the meaning of the verb, we have introduced the possibility that some later process might erroneously identify *A1* as the specific meaning. Note that if the verb sense is never specialized, however, then we cannot get into trouble. But this may be the price we have to pay in order to retain a computationally effective representation. This technique for reducing disjunction in a representation is currently being explored by Etherington et al. (1989).

Other forms of structural ambiguity pose significant problems, although the techniques that I have suggested might be used to advantage. For example, the sentence *The warden reported a fire in the hills* has at least two readings, corresponding to whether the prepositional phrase *in the hills* modifies the act of reporting or the fire itself. One possibility would be a representation where the PP *in the hills* has a uniform interpretation across the potential modifiers, say as a predicate *LOC(X,HILLS),* for some *X*, either the reporting act or the fire, then the representation of this sentence might be (where *W13* is the warden, *F45* is the fire, and *R67* is the reporting act):

$$REPORT(W13,F45,R67) \wedge LOC(X,HILLS) \wedge (X = F45 \vee X = R67)$$

One final technique for ambiguity could involve localized enumeration. If the rest of the interpretation of the sentence is independent of the particular interpretation of a term, that term could be represented by a simple enumeration of its possible interpretations. Completely independent sets of choices do not significantly affect the number of possible interpretations that need to be explicitly constructed. For example, consider a two-place predicate *P* with the first term ambiguous between *A*, *B*, and *C*, and the second ambiguous between *D* and *E*.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

We might write this as $P(\{A,B,C\},\{D,E\})$, where curly brackets indicate an enumerated set of choices. Although there are six possible unambiguous formulas characterized by this term, if the two argument positions are independent of each other, we will never need to enumerate them. Rather, each locally enumerated set can remain until some process reduces its membership to a single term. If the argument positions are not independent, however, we may need to enumerate the cross-product of unambiguous interpretations in order to capture the interactions. For example, if the interactions were such that only four interpretations were possible, we would have to resort to listing explicitly these four alternatives rather than using the single representation. Reasoning systems based on constraint satisfaction (e.g., Waltz, 1975; Allen, 1983) often assume such independence properties. Allen, for example, gives an algorithm for reasoning about temporal intervals that uses all three-way constraints between times, but not higher-level interactions (say interactions between four intervals). Because such higher-level interactions are possible, his algorithm is incomplete, but still performs well in typical everyday situations. To the extent that similar techniques can be developed for disambiguation, local enumeration can be a useful encoding strategy.

6.7 Concluding remarks

Although I have spent most of my time discussing problems with existing approaches, I hope the ideas in the previous section suggest important areas of work in knowledge representation and semantic interpretation. Given the nature of this chapter, I can only come to very broad, vague, conclusions. The central points are that if we are to handle natural language in general, we must take ambiguity seriously, and that we cannot avoid needing knowledge representations that can encode such ambiguities effectively. Furthermore, expressively limited knowledge representations are not going to be of use for natural language systems. As a result, substantial work needs to be done in defining highly expressive representations that have well-understood inferential power. Because complete inference systems for such representations will be intractable, we need more research into ways of defining the inferential power of representations that are not based on completeness properties.

References

Allen, J. (1983). "Maintaining knowledge about temporal intervals", *Comm. of the ACM, 26,* 11, 832–843 (reprinted in Brachman and Levesque, 1985).

Allen, J. F. (1978). *Natural Language Understanding,* Benjamin/Cummings Pub Co.

Allen, J. and Miller, B. (1989). *The RHET User's Manual,* Technical report 238, Dept. of Computer Science, Univ. of Rochester.

Allen, J. and Perrault, C. R. (1980). Analyzing intention in utterances, *Artificial Intelligence, 15,* 3, 143–178.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

Alshawi, H. (1990). "Resolving quasi logical forms", *Computational Linguistics, 16,* 3.

Bar-Hillel, Y. (1971). "Some reflections on the present outlook for high-quality machine translation", in W. P. Lehmann and R. Stachowitz (eds.), *Feasibility Studies on Fully Automatic High Quality Translation,* Technical Report RADC-TR-71-295, Linguistics Research Center, Univ. of Texas at Austin.

Bates, M., Moser, M. G., and Stallard, D. (1986). "The IRUS transportable natural language database interface", in Kershberg, L. (ed.), *Expert Database Systems,* Menlo Park, CA: Benjamin/Cummings Pub. Co.

Bever, T. (1970). "The cognitive basis for linguistic structures", in Hayes, J. (ed.) *Cognition and the Development of Language,* John Wiley, 279–362.

Brachman, R., Fikes, R., and Levesque, H. (1983). A Functional Approach to Knowledge Representation, IEEE Computer 16:10, pp. 67–73.

Brachman, R. and Levesque, H. (eds.) (1985). *Readings in Knowledge Representation,* Morgan-Kaufman.

Brennan, S., Friedman, M., and Pollard, C. (1987). "A centering approach to pronouns", *Proc. of the 25th ACL,* Stanford, CA.

Charniak, E., Gavin, M., and Hendler, J. (1985). *The Frail/NASL Reference Manual,* Technical report CS-83-06, Dept. of Computer Science, Brown University.

Colmerauer, A. (1978). "Metamorphosis Grammars", in L. Bloc (ed.), *Natural Language Communication with Computers,* Berlin: Springer-Verlag.

Cullingford, R. (1981). "SAM", in *Inside Computer Understanding,* Lawrence Erlbaum Associates.

Dyer, M. (1983). *In-depth Understanding,* MIT Press.

Etherington, D., Borgida, A., Brachman, R., and Kautz, H. (1989). "Vivid knowledge and tractable reasoning: preliminary report", *IJCAI-89,* Detroit.

Fawcett, B. and Hirst, G. (1986). "The detection and representation of ambiguities of intension and description", *Proc. of the 24th ACL,* New York.

Frisch, A. (1987). "Inference without chaining", *Proc. IJCAI-87,* Morgan Kaufmann, 515–519.

Grosz, B. J., Appelt, D., Martin, P., and Pereira, F. (1987). "TEAM: An experiment in the design of transportable natural-language interfaces", *Artificial Intelligence, 32,* 2, 173–244.

Grosz, B., Joshi, A., and Weinstein, S. (1983). "Providing a unified account of definite noun phrases in discourse", *Proc. of the Association for Computational Linguistics,* 44–50.

Grosz, B., Sparcek Jones, K., and Webber, B. (eds.). (1986). *Readings in Natural Language Processing,* Morgan Kaufmann.

Hayes, P. J. (1979). "The logic of frames", in Metzing (ed.), *Frame Conceptions and Text Understanding,* de Gruyter.

Hayes, P. (1977). *Some Association-based Techniques for Lexical Disambiguation,* Doctoral Dissertation, Dept. of Mathematics, Ecole Polytechnique federale de Lausanne.

Hirst, G. (1981). *Anaphora in Natural Language Understanding,* Springer-Verlag.

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

Hirst, G. (1990). "Mixed-depth representations for natural language text", *Proc. of the AAAI Symposium on Text-based Intelligent Systems.*

Hobbs, J. (1978). "Resolving pronoun references", *Lingua, 44* (reprinted in Grosz et al., 1986).

Hobbs, J. and Shieber, S. (1987). "An algorithm for generating quantifier scopings", *Computational Linguistics, 13.*

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

Jacobs, P. S. and Rau, L. F. (1990). "SCISOR: A System for Extracting Information from On-line News", *Comm. of the ACM, 33,* 11:88–97.

Katz, J. and Fodor, J. (1963). "The structure of a semantic theory", *Language, 39,* 2:170–210.

Levesque, H. and Brachman, R. (1984). "A Fundamental Tradeoff in Knowledge Representation and Reasoning", in *Proc. of CSCSI-84* (reprinted in Brachman and Levesque, 1985).

McCarthy, J. (1980). "Circumscription: A form of non-monotonic reasoning", *Artificial Intelligence, 13,* 27–39.

McCawley, J. D. (1981). *Everything That Linguists Have Always Wanted to Know about Logic,* U. Chicago Press.

McCord, M. (1986). "Focalizers, the scoping problem, and semantic interpretation", in D. H. Warren and M. van Canegham (eds.), *Logic Programming and its Applications,* Norwood, NJ: Ablex.

Moore, R. C. (1981). "Problems in Logical Form", *Proceedings of the ACL,* 117–124.

Patel-Schneider, P. (1986). "A four-valued semantics for frame-based description languages", *Proc. of the AAAI-86.* Morgan Kaufmann.

Pereira, F. and Warren, D. (1985). "Definite clause grammars for language analysis", *Artificial Intelligence, 25,* 3:301–322.

Perrault, C. R. and Allen, J. (1980). "A plan-based analysis of indirect speech acts", *American Journal of Computational Linguistics, 6,* 167–182.

Quillian, M. R. (1967). "Word concepts: A theory and simulation of some basic semantic capabilities", *Behavioral Science, 12* (reprinted in Brachman and Levesque, 1985).

Reiter, Ray. (1980). "A logic for default reasoning", *Artificial Intelligence, 13,* 81–132.

Ringle, M. (ed.). (1982). *Strategies for Natural Language Processing,* Lawrence Erlbaum Associates.

Scha, R. (1983). *Logical Foundations for Question Answering,* Philips Research Labs, Eindhoven, The Netherlands.

Scha, R. and Stallard, D. (1988). "Multi-level plurals and distributivity", in *Proc. of the 26th ACL.*

Schubert, L. K. (1986). "Are there preference tradeoffs in attachment decisions?" *Proceedings of the AAAI,* 601–605.

Schubert, L. and Hwang, C. H. (1989). "An episodic knowledge representation for narrative texts", *Proc. of the First International Conference on Knowledge Representation,* Morgan Kaufmann.

Schubert, L. and Pelletier, J. (1982). "From English to logic: context-free computation of conventional logical translation", *Computational Linguistics, 8* (reprinted in Grosz et al., 1986).

Searle, J. R. (1975). "A taxonomy of illocutionary acts", in K. Gunderson (ed.), *Language, Mind and Knowledge,* Univ. of Minnesota Press.

Small, S. and Rieger, C. (1982). "Parsing and comprehending with word experts", in Lehnert, W. and Ringle, M., (eds.) *Strategies for Natural Language Processing,* Lawrence Erlbaum Associates, 1982.

Vilain, M. (1985). "The restriction language architecture of a hybrid representation system", *Proc. IJCAI-85,* Morgan Kaufmann, 547–554.

Vilain, M. and Kautz, H. (1986). "Constraint propagation algorithms for temporal reasoning", *Proc. of the AAAI,* 377–382.

Waltz, D. (1975). "Understanding line drawings of scenes with shadows", in P. Winston (ed.), *The Psychology of Human Vision,* McGraw-Hill.

Weischedel, R. (1989). "A hybrid approach to representation in the JANUS natural language processor", *Proc. of the 27th Meeting of the ACL,* 193–202.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

Wilensky, R. (1983). *Planning and Understanding*, Addison-Wesley.

Wilks, Y. (1975). "An intelligent analyzer and understander of English", *Artificial Intelligence, 6*, 264–274.

Woods, W. A. (1975). "What's in a link?", in D. Bobrow and A. Collins (eds.), *Representation and Understanding*, Academic Press.

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press

https://doi.org/10.1017/CBO9780511659478.008 Published online by Cambridge University Press