Underspecification in Semantic Representations: Which Technique for What Purpose?

Harry Bunt
Dep. of Computational Linguistics and Art. Intelligence, Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands
bunt@uvt.nl

Abstract

This paper examines a number of techniques for underspecification in semantic representations, notably holes and labels, ambiguous constants, metavariables, radical reification, stores, lists and disjunctions, and *in situ* quantified terms. These techniques are considered for their usefulness in dealing with a variety of linguistic phenomena and cases of incomplete input, which have motivated the use of underspecified semantic representations. It is argued that holes and labels, and the use of ambiguous constants and variables, have nearly disjoint domains of application, and together cover a wide range of phenomena.

1 Introduction

In the last ten years, a variety of representation formalisms have sprung up that support underspecified semantic representation, and that in some cases have been designed specifically for that purpose. These include Quasi-Logical Form (QLF, Alshawi, 1990); Underspecified Logical Form (ULF, Geurts & Rentier, 1991); Underspecified Discourse Representation Theory (UDRT, Reyle, 1993); Minimal Recursion Semantics (MRS, Copestake et al., 1995); Ontological Promiscuity (Hobbs, 1985); Hole Semantics (Bos, 1995). These formalisms support underspecification techniques which are sometimes rather different, and sometimes quite similar but uses different notations and terminology. For example, Pinkal (1999) uses the term 'metavariable' as referring to the 'labels' of UDRT, the 'handles' of MRS, and the 'holes' of Hole Semantics, while other authors have used 'metavariable' to refer to ambiguous terms. This paper aims at starting a systematic discussion of what the various underspecification techniques have in common and in what respects they differ. We do this by considering what different techniques have to offer for dealing with a number of phenomena that motivate the use of underspecification.

2 Why underspecified semantic representation

The interest in underspecified semantic representations comes from attempts to design efficient language understanding systems. Ambiguity is a notorious bottleneck for such systems. Two phenomena that are particularly important in this context are the pervasive ambiguity of words, and the large number of readings that correspond to the possible relative scopings of quantifiers.

The efficiency of language interpretation can be increased greatly by postponing to distinguish semantically different but syntactically identical readings of lexical items to later stages of processing, and in general by postponing to distinguish alternative readings of words and phrases which require context information for disambiguation.

An independent motivation for assigning underspecified semantic representations (USRs) to sentences comes from machine translation, since ambiguities in a sentence in the source language are often retained in the target language. Speech input provides another motivation: imperfect recognition may lead to having to deal with incomplete input. The occurrence of unknown words (e.g., proper names) is similar to incomplete input. Finally, psycholinguistic considerations also provide arguments in favour of USRs: when listening to a sentence, people construct interpretations incrementally; this means that semantic representation as are constructed from incomplete input.

In the following sections we will discuss the applicability of a variety of underspecification techniques to various important forms of lexical, syntactic, semantic, and discourse ambiguity, and to cases of incomplete input information. Limitations of space force us to be brief and in no way exhaustive; moreover, in some cases more thorough analysis is needed. We nevertheless hope that this is a useful step in determining the possibilities and limitations of the underspecification techniques that have been proposed. We summarize our findings in Table 2.

3 Underspecification techniques

Semantic representations have traditionally been cast in formal languages defined in mathematical logic. In such languages, an expression is formed by combining subexpressions by means of constructions like function application, conjunction, and existential quantification; each of these constructions has a semantic definition that specifies how the denotation of a constructed expression can be computed from the denotations of its subexpressions. Since the atomic subexpressions (constants and variables) have precisely specified denotations, it follows that the meaning of every complex expression is also precisely defined.

A semantic representation can be underspecified in two ways:[1]

1. Constants and variables may be ambiguous or vague;
2. The way in which subexpressions are combined by means of constructions may not be fully specified.

A representation which is underspecified in one or both of these ways may be viewed as a representation of *constraints* on meaning representations, i.e. as a meta-representation describing the set of representations that satisfy the constraints (cf. Muskens, 2001). An underspecified semantic representation of a sentence is therefore a pair USR $= < E_U, C_U >$, where E_U is a set of (sub-)expressions, representing the meanings of parts of the sentence, possibly containing ambiguous constants and variables, and C_U is a set of constraints: (a) on the admissible ways of combining subexpressions; (b) on the possible ways of 'instantiating' ambiguous constants and variables.[2] Therefore, USRs require not only a language for semantic representation, but also a constraint language for specifying constraints on underspecified elements. The various frameworks that have been proposed for USRs differ in both languages.

The most important techniques for semantic underspecification that have been proposed can be classified as follows.

[1] In theory, a third way to introduce underspecification could be to allow ambiguity in constructions. For example, one might allow a two-place predicate to combine with a set rather than an ordered pair of two arguments, leaving the role of each of the arguments underspecified. We are not aware of any proposal in this direction.

[2] This is the standard notion of underspecified semantic representation. In principle, USRs could be defined recursively, E_U being a set of USRs.

1. **Ambiguous or vague terms** in the representation language. The following cases may be distinguished:

    (a) Predicate constants, function constants or individual constants that represent ambiguous or vague referring expressions in natural language. Examples are predicate constants that represent homonymous or polysemous nouns and verbs. We will refer to these as *referential metaconstants.*

    (b) Constants that represent a semantic relation which is not expressed in natural language, such as the relation between the nouns in a nominal compound (*toxic waste dump*), or the relation between a noun and a verb in a metonymical expression like *The office called.* We will refer to such terms as *formal metaconstants.*

    (c) Individual or other variables which are intended to be replaced by a constant or a bound variable that occurs elsewhere in the representation, as in ellipsis. We will refer to such terms as *metavariables.*

2. **Labels and holes.** For underspecifying the way in which the E_U-elements in a USR $U =< E_U, C_U >$ are to be combined, a *label* may be associated with each element in E_U, and the fact that a certain subexpression labelled L_1, consists of two subexpressions joined by means of the construction κ, of which the first is the subexpression labelled L_2 but the second is unknown, may be represented as $L_1 : \kappa(L_2, h_1)$, where h_1 is a variable that ranges over E_U. Following Bos (1995), such variables are called *holes.*

    The labeling of subexpressions and the use of holes to refer to subexpressions in scope constraints in a metalanguage, is clearly something that can be applied to any given object language. Bos (1995) formalizes the use of labels and holes in propositional and (dynamic) predicate logic in a way that is easily extended to other object languages, and may be credited to have defined 'Hole semantics'.

3. **Radical reification.** The davidsonian approach of reifying states and events has become standard in modern semantics. Hobbs (1983) has proposed to apply reification not only to verbs but also to nouns, adjectives, adverbs, and prepositions, constructing 'flat' representations in the form of conjunctions in first-order logic. The bewildering

ontological picture behind this idea has made Hobbs refer to his approach as 'ontological promiscuity'. We will call this approach *radical reification* (RR).

4. **In situ representation** of scoped elements. One of the oldest documented attempts to represent quantifier scopes in an underspecified way[3] is that of Alshawi's Quasi-Logical Form (QLF), where predicates do not have arguments denoting individuals, but have quantified arguments. So essentially a sentence like *Every student reads two books* is represented as (read (every student, two books)).

5. **Stores and lists.** A relatively old idea is to accompany the construction of semantic representations by a symbolic memory in which those components of a representation are temporarily stored of which the place in the representation is not yet fully determined. Cooper storage is a technique that was developed for doing this for quantifier scopes (Cooper, 1983). Keller (1998) has developed an improved version of the Cooper store techniques with nested stores, known as Keller stores. Some experimental language understanding systems use a similar technique, placing NP representations on a list from which they can be tretrieved in an order that corresponds to their relative scopes.

   In ULF, lists of variables are used as an alternative to metavariables (in the sense defined above) constrained to be instantiated as one of the elements in the list. Also, lists of predicates are used to indicate that the predicates are to be combined somehow in order to form a complex predicate.

4 Applicability of underspecification techniques

4.1 A taxonomy of sources of underspecification

In Table 1 we provide an overview of various sources of ambiguity, vagueness, or lack of information, whose treatment has motivated the development of underspecification techniques. The table lists a number of instances that are particularly relevant in this respect. We will subsequently analyse a number of these for the applicability of the underspecification techniques mentioned

[3] The oldest way appears to be the one by Woods in the LUNAR question-answering system. See Woods, 1978

in the previous section. These techniques have typically been developed within a representation formalism, such as QLF or MRS; these formalisms do not all have equally solid logical foundations. However, König & Reyle (1996) have developed a logical basis for a broad range of underspecification formalisms.

Source of underspecification	*Instance*
Lexical ambiguity	homonymy polysemy anaphora deixis count/mass
Structural semantic ambiguity	quantifier scope quantifier distributivity modifier distributivity nominal compounding metonymy
Syntactic ambiguity	modifier attachment relative clause attachment syntactic scope thematic role
Discourse ambiguity	ellipsis short answers anaphora, cataphora
Missing information	unknown words incomplete input incremental processing

Table 1. Sources of semantic underspecification.

4.2 Lexical ambiguity

4.2.1 Homonomy and polysemy

The efficient processing of sentences with homonymous or polysemous words has motivated the oldest known use of underspecified semantic representation languages. In the PHLIQA question answering system (see Bronnenberg et al., 1979) a semantic representation language was used with a *referential metaconstant* for every English content word, the idea being that there is hardly any content word that does not display some degree of homonymy or polysemy. So for instance, the metaconstant 'AMERICAN' was used to represent the various senses of the word *American* as illustrated by *American car* and *American city*. A domain-specific lexicon listed the possible instantiations of each metaconstant, such as AMERICAN having as one of its instantions: $\lambda x.\ \mathsf{Country\text{-}of}(\mathsf{Manufacturer\text{-}of}(x)) = \mathsf{USA}$.

One may try to deal with clear cases of homonymy by means of alternative lexical entries, using holes and labels to represent the alternatives in an USR, but polysemy and lexical vagueness in general cannot be handled by such an approach, which presupposes a definite, finite number of alternatives. Referential metaconstants seem the perfect instrument for underspecifying these forms of semantic indeterminacy.

4.2.2 Anaphora and deixis

Pronouns and definite NPs can be ambiguous between anaphoric and deictic use (*Did you see that?*); in either case, their intended referent has to be found in the context, or to be inferred from it, which adds another level of ambiguity.The simplest way to underspecify the intended interpretation of an anaphoric or deictic expression would seem to be the use of a metavariable, with the syntactic/semantic constraints that the expression provides. For instance, for the English pronoun *her* as occurring in *Did you see her* the constraints include that the referent plays the role of the theme in a *see* event and is a female person or animal. This underspecification technique has been used in the DENK system (see Kievit et al, 2001).

4.2.3 Count/mass ambiguities

Examples such as *There's no chicken in the yard* and *There's no chicken in the salad* illustrate the count/mass ambiguity that is found in many languages. In English, virtually every noun can be used both as a count noun and as a mass noun.[4] Treating every noun in the lexicon as ambiguous between a count and a mass reading is computationally unattractive and conceptually unsatisfactory, since there are systematic semantic relations between the count and mass uses of a word. Bunt (1985) has introduced the use of *formal metaconstants* for converting a count noun to its mass use and the other way round. For instance, the noun *bread* is represented in the lexicon as BREAD, and the grammar rules for constructing noun phrases may turn this into μ(BREAD), adding a function constant μ that can be instantiated in two ways. This approach has been implemented in the TENDUM dialogue system (Bunt et al., 1984).

4.3 Structural semantic ambiguity

4.3.1 Quantifier scope

Underspecified semantic representation of quantifier scopes have been proposed using the following of the above mentioned techniques.

1. In situ representation. Alshawi's original QLF proposal, implemented in the Core Language Engine (CLE), uses indexed quantified terms ("qterms") as predicate arguments, leaving the relative scopes of the NPs unspecified. This representation has been used in the Core Language Engine, in combination with a disambiguation process that pulls the quantifiers out of the predicate arguments and assigns them a scope (Alshawi, 1990).

Willis & Manandhar (2001) have argued that QLF does not represent *partial* scope information adequately (for which the indices of the qterms can be used), and for instance does not get the scope constraints for the sentence *Every representative of a company saw some samples* right. Pulman (2000) also notes shortcomings of the CLE QLF and introduces an improved version of QLF. Other forms of in situ representation are those in Allen's textbook and those used in ULF (Geurts & Rentier, 1991; Kievit, 1994), that have been implemented in the PLUS system.

[4] For instance, in a pamphlet in a hospital the following text was found: *Nurses spend a great deal of their time washing patients, and since the population has become more than 10% taller in the last ten years, they have correspondingly more patient to wash.* Conversely, mass nouns often allow a count use (*Two coffee, please; That's a great wine*).

2. Quantifier storage. The mechanism of Cooper storage (and Keller storage) is not an underspecified semantic representation technique as such, but a formal procedure for postponing to make a decision on the relative scopes of noun phrases during much of the interpretation of a sentence. Some language understanding systems of the seventies and eighties, such as PHLIQA, SPICOS and TENDUM have incorporated similar ideas.

Bunt and Muskens (1999) have described a formal calculus assigning logical forms to syntactic trees using an NP store, which (like Keller store) has the nice property that when an NP can raise out of an NP, as in *Every representative of a company saw some samples*, this is allowed only if the embedded NP is retrieved from the store later than the embedding one.

3. Metaconstants.
A treatment with formal metaconstants has been implemented in the TENDUM dialogue system. All the noun phrases in a clause are collected in a 'noun phrase sequence' constituent, collecting the NP representations as the argument of a metafunction. Alternative instantiations of this metafunction correspond to alternative relative scopings. Clause formation occurs through combination of the NP sequence with the verbal constituent. A shortcoming of this technique is that sentence-specific constraints on the possible instantiations of the metaconstant cannot be expressed as constraints on metaconstant instantiation, hence there is no adequate representation of partial scoping.

4. Labels and holes. The use of labels with scope constraints was invented in DRT, leading to UDRT, in particular for the underspecified represention of quantifier scopes. In MRS the use of labels is applied to attribute-value matrices, fitting in with the object language of HPSG. Willis & Manandhar (2001) argue that UDRT cannot represent partial scoping adequately. We believe that this should be possible, however, if a suifficiently powerful constraint language is used. The ideas of UDRT and MRS have been implemented successfully in the Verbmobil system (see Schiehlen, Bos & Dorna, 2000).

5. Radical reification.
Hobbs (1983) has proposed 'flat', conjunctive semantic representations which contain less information then is usually considered adequate, but where an abduction-driven pragmatic component in the interpretation process supplies additional information. The flatness of the representations is attractive,

but the price is a rather cumbersome treatment of quantification, involving the notion of a 'typical element' of every set, and other ontologically strange creatures. For instance, the sentence *Most men work* would be represented in RR as:

(1) $(\exists s_2, s_1, x, e, y)[\mathsf{most}(s_2, s_1) \wedge \mathsf{dset}(s_1, x, e) \wedge \mathsf{man}(e, x) \wedge \mathsf{typelt}(y, s_2) \wedge \mathsf{work}(y)]$

which is to be read as: There is a set s_1 defined by the property e of its typical element x being a man, and there is a set s_2 which is most of s_1 and has y as its typical element, and y works.

Technically, RR has the same shortcoming as several other proposals of not being able to express partial scopings, since this cannot be expressed in terms of constraints on reification variables.

4.3.2 Quantifier distributivity

The ambiguities involved in the *distributivity* of quantifiers has received much less attention than those in quantifier scoping, although every NP introduces distributivity ambiguities. A single NP can be argued to be multiply ambiguous between individual, collective, and group readings, and the combination of NPs gives rise to additional distributivity ambiguities. Some examples are:

(2) a. These machines will lift the platform. [together]

b. These machines lift 5 crates. [in one go]

c. These machines have lifted 2000 crates. [in total/each]

The only approach to underspecifying quantifier distributivity that has been proposed, to our knowledge, is the use of formal metaconstants (Bunt, 1985). In this approach, predicates like Lift are applied to arguments from a domain represented as $\delta(NUM, \mathsf{MACHINES})$, where δ is a formal metaconstant representing distributivity, and NUM stands for the (absolute or relative) numerical information in the (generalized) quantifier. The metaconstant can be instantiated as $(\lambda M.\ \lambda X.\ X)$, as $(\lambda M.\ \lambda X.\ \{X\})$, and as $(\lambda M.\ \lambda X.\ \{Y|Y \subseteq X \wedge NUM(Y)\})$, for individual, collective, and group dstributivity (with group size NUM), respectively.

4.3.3 Modifier distributivity

Distributivity ambiguities arise not only in quantification but also in modification, as in *The crates that this machine lifted*, which can be taken both individually and collectively. The kind of representation of quantifier distributivity that we have just seen, by means of formal metaconstants, can also be used to represent modifier distributivity in an underspecified way

4.3.4 Nominal compounds

For handling nominal compounds, Hobbs et al. (1993) have proposed a treatment with formal metaconstants, to be instantiated through abductive reasoning with context information. For example, the compound *Boston office* in the sentence *The Boston office called* is represented schematically as $\mathsf{boston}(x) \wedge \mathsf{office}(x) \wedge \mathsf{NN}(x)$, where NN is a metaconstant representing the unknown semantic relation between the two nouns.

The interpretation problem for nominal compounds is very similar to that of metonymy (in fact, *The Boston office called* is also metonymous). We will next discuss Pinkal's suggestion to treat metonymy with the help of labels and holes; what is said there applies equally to nominal compounds.

4.3.5 Metonymy

Metonymy is one of the types of ambiguity for which Pinkal (1999) suggests an underspecified treatment with holes and labels. For the example sentence *John began the book* he provides the schematic representation (3a).

(3) a. $< \{X_0\text{: }\mathsf{begin(john,}\ X_1\mathsf{)},\ X_2\text{: }\mathsf{the\text{-}book}\}, \{X_1 > X_2, X_0 > X_1\ \} >$

b. $\mathsf{begin(john,\ writing\text{-}of(the\text{-}book))}$

The subexpressions labelled X_0 and X_2 in this representation are not connected, but *the book* is constrained to be outscoped by the unspecified subexpression labelled X_1. The idea is of course that X_1 identifies a subexpression that should be plugged in and connect the two subexpressions, such as X_1: $\mathsf{writing\text{-}of}(X_2)$, giving the result (3b), after replacing labels by the subexpressions that they label and suppressing the top label X_0.

It should be noted, however, that the use of labels and holes that we see here is not in accordance with the definitions of Hole Semantics. In Hole Semantics labels are introduced only in order to formulate constraints on the possible instantiation of holes, which range over the labels in the E_U part of a USR $< E_U, C_U >$. But a hole like X_2 in the above example does not relate to any element in the set $E_U = \{X_0$: begin(john, X_1), X_2: the-book$\}$. Instead, X_2 stands for any object-level expression that can be generated through reasoning with contextual information.

The use of a term that stands for an expression which is not given in the underspecified representation, but that has to be generated from the (nonlinguistic) context, is is in fact the use of a *metaconstant*.

4.4 Syntactic ambiguity

Of the many forms of syntactic ambiguity, we consider syntactic scope ambiguities and attachment ambiguities.

4.4.1 Syntactic scope ambiguities

Syntactic scope ambiguities can be handled by means of labels and holes. For instance, Bos (1995) shows how the sentence (4a) can be represented schematically without resolving the relative scopes of *do not* and *and* by the USR (4b).

(4) a. Do not sleep and pay attention, please.

b. $< \{L_1 : \neg h_1,\ L_3 : \mathsf{sleep},\ L_4 : \mathsf{pay\text{-}attention},\ L_2 : h_2 \wedge h_3\},$
$\{h_1 \geq L_3, h_2 \geq L_3, h_3 \geq L_4\} >$

The constraints in the second part of the USR express that the argument of the negation outscopes *sleep* and that the two arguments of *and* outscope *sleep* and *pay attention*, respectively.

Molla (2001) has proposed a variant of radical reification in which all logical relations are reified as well, resulting in flat list representations. For the most plausible reading of (4a) the RR representation would be:

(5) $\mathsf{not}(e_1, e_3),\ \mathsf{sleep}(e_3, x),\ \mathsf{and}(e_4, e_1, e_6),\ \mathsf{pay\text{-}attention}(e_6, x)$

To underspecify the scoping, the arguments of the scope-bearing elements are simply not tied together, and we get the USR (6):

(6) $<\{\mathsf{not}(e_1, e_2),\ \mathsf{sleep}(e_3, x),\ \mathsf{and}(e_4, e_5, e_6),\ \mathsf{pay\text{-}attention}(e_6, x)\},$
$\{(e_2 = e_3 \wedge e_5 = e_1) \vee (e_2 = e_4 \wedge e_5 = e_3)\} >$

Unfortunately, the variant of RR that Molla proposes suffers from technical problems in dealing with quantification that seem to be insuperable.

4.4.2 Attachment ambiguities

An attachment ambiguity occurs when a sentence contains several candidates for being modified by a certain modifier. Two important cases of this are PP attachment and relative clause attachment, as illustrated by:

(7) a. John saw the man with binoculars.

b. The crates on the platform that Hercules lifted.

Syntactically, the different ways of attaching the PP and the relative clause come down to different ways of connecting the subtree, describing the modifier, to the rest of the tree, and it might seem that this corresponds semantically to different ways of inserting the representation of the modifier in the rest of the semantic representation. Underspecifying the attachment would then take the form of keeping the modifier representation separate and indicating its possible insertion points. Labels and holes would seem to be the obvious instruments for achieving this. Schematically, we can represent the two readings of (7a) as (8a) and (8b), and in labelled form as (9a) and (9b).

(8) a. $\mathsf{saw}(e_1,\mathsf{j},\mathsf{x}) \wedge \mathsf{theman}(\mathsf{x}) \wedge \mathsf{withbinocs}(e_1)$

b. $\mathsf{saw}(e_1,\mathsf{j},\mathsf{x}) \wedge \mathsf{theman}(\mathsf{x}) \wedge \mathsf{withbinocs}(\mathsf{x})$

(9) a. $\{L1 : \mathsf{saw}(e_1,\mathsf{j},\mathsf{x}),\ L2{:}\mathsf{theman}(\mathsf{x}),\ L3{:}\mathsf{withbinocs}(e_1)\}$

b. $\{L1 : \mathsf{saw}(e_1,\mathsf{j},\mathsf{x}),\ L2{:}\mathsf{theman}(\mathsf{x}),\ L3{:}\mathsf{withbinocs}(x)\}$

The only difference between the two readings is the argument of the modifier. Indeed, from a semantic point of view an attachment ambiguity is a choice of modifier argument. Therefore, holes can be used for underspecifying the attachment only if the alternative arguments of the modifier are labelled subexpressions, as in (10):

(10) $< \{L1 : \mathsf{john},\ L2 : e_1,\ L3 : \mathsf{saw}(L2, L1, L3),\ L4 : \mathsf{theman}(L5),\ L5 : x,$
$L6 : \mathsf{withbinocs}(h_1)\},\ \{h_1 = L2 \vee h_1 = L5\} >$

Technically this seems possible, but it does not make sense to label a variable or a constant, as in $L2 : e_1$ and $L5 : x$, since the constant and variable themselves can be inserted in the semantic representation, obtaining the simpler USR (11).

(11) $< \{\mathsf{saw}(e_1,\mathsf{j},\mathsf{x}) \wedge \mathsf{theman}(\mathsf{x}) \wedge \mathsf{withbinocs}(h_1)\},\ \{h_1 = e_1 \vee h_1 = x\} >$

In this representation, h_1 is clearly a variable that can be instantiated as a constant or an object-language variable, hence h_1 is in fact not a hole but a metavariable.

4.5 Discourse ambiguity

Of the various kinds of discourse ambiguity, we consider ellipsis here. Short answers are similar. For anaphora see 4.2.2.

4.5.1 Ellipsis

In the case of intrasentential ellipsis, some linguistic material is absent which can be supplied from elsewhere in the sentence. This seems an ideal application of labels and holes, at least in straightforward cases of ellipsis. Pinkal (1999) points out that parallelism has to be taken into account, however, for instance to make sure that the relative scope assignments in the first and the second part of the sentence *Two European languages are spoken by every linguist, and two Asian languages are, too* are the same. So the constraints in the USR should take such parallelism into account. Pinkal refers to a way in which this has been accomplished in the ücken CHORUS project. Adequate underspecified representations of ellipsis by means of labels and holes are thus possible, provided that a more powerful constraint language is used than is common in Hole Semantics.

4.6 Incomplete information

4.6.1 Unknown words

The occurrence of unknown words in the input to a language understanding system may be considered as causing ambiguity or semantic vagueness in the extreme: unknown words can mean anything that is appropriate in the context of utterance. Therefore the treatment of ambiguous and vague words outlined above by means of referential metaconstants, can also be applied in this case.

4.6.2 Incomplete input and incremental processing

Incomplete input and incremental processing both have the effect that the utterance interpretation process has the task of assigning a semantic representation to an incomplete fragment of the input.

Pinkal (1999) suggests to treat the ambiguities arising because of incomplete input by means of labels and holes. For the example sentence *We meet XX next week*, where *XX* marks an unrecognized part of the input, he suggests the (schematic) underspecified representation (12).

(12) $< \{X_1$: meet(we), X_2: next-week$\}$, $\{X_0 \geq X_1, X_0 \geq X_2\} >$

where the two semantic chunks labelled X_1 and X_2 might get linked up "by adding the constraints $X_3 = T\_rel(X_2)$ and $X_0 = X_3(X_1)$, where T_rel is a meta-variable ranging over temporal relations" and X_0 is the top label (*Op. cit.*, p. 37). The result of taking these constraints into account would be that we obtain a fully specifed representation like (13).

(13) (In(next-week))(meet(we)

While this seems OK as such, notice that in the process (a) we have used a constraint of the form $X_3 = T\_rel(X_2)$, which is not a constraint of the kind that Hole Semantics allows but which contains a semantic construction of the *object* language, as well as a referential metaconstant T_rel ranging over object-language predicates; and (b) the additional constraints introduce a new label X_3, which does not occur in the original USR and which does not range over USR subexpressions. Therefore, this is again not really an application of the technique of holes and labels; instead, the variable X_3 is a metaconstant.

5 Summary and conclusion

We summarize the applicability of the various underspecification techniques in Table 2 below, where ± indicates applicability if the technique in question is supplemented with an adequate constraint language. We see that each underspecification technique has its own specific areas of application and that, except for some cases of syntactic ambiguity, the various techniques complement rather than compete with each other. It is also worth noting the the techniques of holes and labels together with that of metaconstants and -variables provides full coverage of all the cases of ambiguity, vagueness, and incomplete information considered here. A representation language that incorporates both these techniques would therefore seem ideally suited for underspecified semantic representation.

	Labels and holes	Meta-variables	Refer'al meta-cons.	Formal meta-cons.	Radical reification	Stores, lists, in situ
Lexical ambiguity						
homonymy	+	–	+	–	–	–
polysemy	–	–	+	–	–	–
anaphora	–	–	–	+	–	–
deixis	–	–	–	+	–	–
count/mass	–	–	–	+	–	–
Struct. semantic ambiguity						
quantifier scope	+	–	–	–	–	±
quantifier distributivity	–	–	–	+	–	–
modifier distributivity	–	–	–	+	–	–
nominal compounding	–	–	–	+	–	–
metonymy	–	–	–	+	–	–
Syntactic ambiguity						
modifier attachment	–	+	–	–	±	–
syntactic scope	+	–	–	–	±	–
thematic role	–	+	–	+	±	–
Discourse ambiguity						
ellipsis	±	–	–	–	–	–
short answers	+	–	–	–	–	–
anaphora and cataphora	–	–	–	+	–	–
Missing information						
unknown words	–	–	+	–	–	–
incomplete input	–	–	–	+	–	–
incremental processing	–	–	–	+	–	–

Table 2. Applicability of underspecification techniques.

References

Alshawi, H. (1990) Resolving Quasi Logical Form. *Computational Linguistics* 16:133–144.

Bos, J. (1995) Predicate Logic Unplugged. In *Proceedings of the 10th Amsterdam Colloquium*, Amsterdam: ILLC, 133–142.

Bronnenberg, W., H. Bunt, J. Landsbergen, R. Scha, W. Schoenmakers & E. van Utteren (1979) The question answering system PHLIQA1. In L. Bolc (ed.) *Natural language question answering systems.* London: McMillan, 217–305.

Bunt, H. (1984) The resolution of quantificational ambiguity in the TENDUM system. *Proceedings COLING 1984*, Stanford University, 130 – 133.

Bunt, H. (1985) Mass terms and model-theoretic semantics. Cambridge University Press.

Bunt, H., R.J. Beun, F. Dols, J.v.d. Linden & G. Schwartzenberg (1984) The TENDUM dialogue system and its theoretical foundations. *IPO Annual Progress Report* 19, 105–113, Eindhoven: IPO.

Bunt, H. & R. Muskens (1999) Computational Semantics. In H. Bunt & R. Muskens (eds.) *Computing Meaning, Vol. 1.* Dordrecht: Kluwer, 1–32

Cooper, R. (1983) *Quantification and Syntactic Theory.* Dordrecht: Reidel.

Copestake, A., D. Flickinger, R. Malouf, I. Sag & S. Riehemann (1995) Minimal Recursion Semantics. Unpublished ms., CSLI, Stanford University.

Geurts, B. & G. Rentier (1991) Quasi logical form in PLUS. Esprit project P5254 (PLUS) internal report, ITK, Tilburg,

Hobbs, J. (1985) Ontological Promiscuity. In *Proc. 23rd Annual meeting of the ACL*, Chicago, 61 – 69.

Hobbs, J., M. Stickel, D. Appelt & P. Martin (1993) Interpretation as Abduction. *Artificial Intelligence* 63: 69 – 142.

Keller, W. (1998) Nested Cooper Storage: The Proper Treatment of Quantification in Ordinary Noun Phrases. In: U. Reyle & C. Rohrer (eds.) *Natural Language Parsing and Linguistic theories.* Dordrecht: Reidel, 1–32.

Kievit, L. (1994) *Context-driven natural language interpretation.* Ph.D. Thesis, Tilburg University.

Kievit, L., P. Piwek, R.J. Beun & H. Bunt (2001) Multimodal Cooperative Resolution of Referential Expressions in the DenK system. In H. Bunt & R.J. Beun (eds) *Cooperative Multimodal Communication.* Heidelberg: Springer, 197–214.

König, E. & Reyle, U. (1996) A General Reasoning Scheme for Underspecified Representations. In: H.J. Ohlbach & U. reyle (eds.) *Logic and its Applications. Festschrift for Dov Gabbay.* Dordrecht, Kluwer, 1–28.

Molla, D. (1999) Ontologically Promiscuous Flat Logical Forms for NLP. In *Proc. 4th Intern. Workshop on Computational Semantics*, Tilburg Univ., 249 – 265.

Muskens, R. (2001) Talking about Trees and Truth-Conditions. *Journal of Logic, Language and Information* 10: 417–455.

Pinkal, M. (1999) On semantic underspecification. In H. Bunt & R. Muskens (eds.) *Computing Meaning, Vol. 1.* Dordrecht: Kluwer, 33–55.

Pulman, S. (2000) Bidirectional Contextual resolution. *Computational Linguistics* 26:4. 497–538.

Reyle, U. (1993) Dealing with Ambiguities by Underspecification. *Journal of Semantics* 10 (2), 123–179.

Schiehlen, M., Bos, J. & Dorna, M. (2000) Verbmobil Interface Terms (VITs). In W. Wahlster (ed.) *Verbmobil: Foundations of Speech-to-Speech Translation.* Berlin: Springer, 183–199.

Willis, A. & S. Manandhar (2001) The Availability of Partial Scopings in an Underspecified Semantic Representation. In H. Bunt, R. Muskens & E. Thijsse (eds.) *Computing Meaning, Vol. 2*, Dordrecht: Kluwer, 129 – 145.

Woods, W. (1978) Semantics and quantification in question answering. In M. Yovits (ed.) *Advances in Computers.* New York: Academic Press, 2–64.