Semantic Underspecification: Introduction

Laura Kallmeyer
Heinrich-Heine-Universität Düsseldorf

Wintersemester 2011/2012

Overview

1. Introduction
2. Approaches to Underspecification
3. Describing Ambiguity
4. Deriving Ambiguity
5. Processing Underspecified Representations

[Egg, 2010]

Introduction (1)

What is underspecification?

- Underspecification can be defined as the deliberate omission of information from linguistic descriptions to capture several alternative realisations of a linguistic phenomenon in one single representation.
- Underspecification emerged in phonology and was later adopted by semanticists to model ambiguity.
- Underspecified semantic representations capture whole sets of different meanings (one for each reading of an ambiguous expression) in one representation.
- Semantic underspecification focusses on expressions with systematically related sets of readings, in particular, on scope ambiguity.

Introduction (2)

Where is underspecification used?

- In natural language processing, underspecification is endorsed to keep semantic representations of ambiguous expressions tractable and to avoid unnecessary disambiguation steps.
- A new use of underspecification is its use in hybrid processing, where it is used as a common format for the results of deep and shallow processing.
- Underspecification is used also in syntax and discourse analysis.

Approaches to underspecification (1)

Data: quantifier scope ambiguities.

(1) Every woman loves a man.
a. $\forall > \exists$: $\forall x(woman'(x) \rightarrow \exists y(man'(y) \wedge love'(x,y)))$
b. $\exists > \forall$: $\exists y(man'(y) \wedge \forall x(woman'(x) \rightarrow love'(x,y)))$

- The formulae in (1a.) and (1b.) consist of the same three parts (roughly, the semantic contributions of the verb and its two arguments), and the relation *love'* introduced by the verb always gets lowest scope.
- The formulae only differ in the arrangement of the semantic contributions of the arguments of the verb.

Approaches to underspecification (2)

More difficult cases of nested quantification:

(2) Every researcher of a company saw most samples

- The challenge of nested quantification is the fact that the number of readings is less than the number of the possible permutations of its quantifiers w.r.t. their scope ordering.
- In (2), there are $3! = 6$ possible permutations but at least the scope ordering $\forall > most' > \exists$ is not attested [Hobbs and Shieber, 1987]). $\exists > most' > \forall$ seems not possible neither [Joshi et al., 2003, Kallmeyer and Romero, 2008].

Approaches to underspecification (3)

- Appropriate underspecification formalisms must be able to represent the exact range of readings of an ambiguous expression and may not overgenerate by predicting unattested readings. This is accomplished in two ways.
- First, ambiguity can be described: Expressions of a formalism describe the set of readings of an ambiguous expression so closely that this suffices to determine the range of its readings. Procedures that derive the individual readings then merely enumerate the readings, they do not restrict them in any way.
- Second, ambiguity can be derived: Some formalisms provide an initial, more general characterisation of the readings; the exact range of readings is then only determined by specifying a procedure (an algorithm) to derive fully specified readings from the general characterisation.

Describing Ambiguity (1)

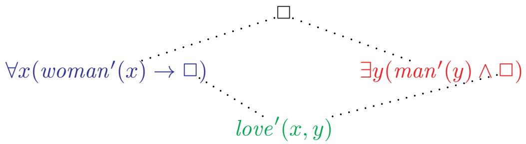
- Use partial descriptions for the sets of semantic representations for the readings of ambiguous expressions.
- Requires that these sets can be characterised by a property/a description that exclusively holds for their elements. This description characterizes the common ground between the semantic representations only.
- Most underspecification formalisms that follow this strategy distinguish an object level (semantic representations) and a meta level (descriptions of these representations, called *constraints*).

Describing Ambiguity (2)

(3) Every woman loves a man.

a. $\forall x(woman'(x) \rightarrow \exists y(man'(y) \wedge love'(x,y)))$

b. $\exists y(man'(y) \wedge \forall x(woman'(x) \rightarrow love'(x,y)))$

(4) 

(4) comprises four fragments of semantic representations containing holes (indicated by □). The dotted lines indicate *part-of* or *sub-structure* or *scope* relations. This relation is transitive.

Describing Ambiguity (3)

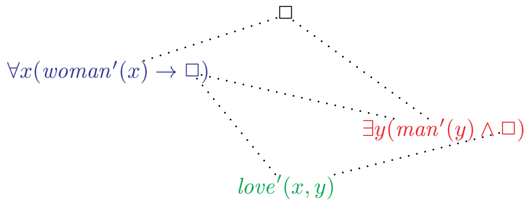
(4) can be paraphrased as follows:

- The fragment at the top is just a hole, i.e. the described representations are not yet known.
- This topmost hole contains (outscopes) both quantifiers.
- Finally, the holes in both the right and the left quantifying fragment are related to the bottom fragment in terms of the outscoping relation, i.e., the bottom fragment is in the scope of either quantifier.

The only semantic representations compatible with this description are the two in (3), as desired.

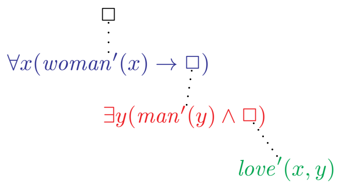
Describing Ambiguity (4)

To derive the described readings from the constraints, we extend the scope relation until the scope of all fragments is fixed. For (4a.), this yields:

(5) 

Describing Ambiguity (5)

Now we omit all scope relations that already follow from the transitivity of scope.

(6) 

Pairwise identification of the hole-fragment tuples yields reading (4a.).

Describing Ambiguity (6)

Underspecification formalisms that implement scope in this way:

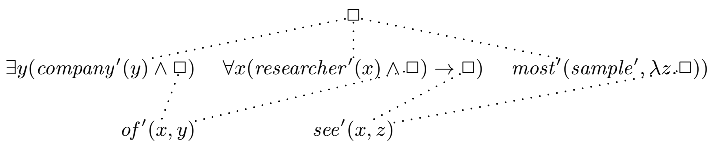
- Underspecified Discourse Representation Theory [Reyle, 1993, Reyle, 1996]
- Minimal Recursion Semantics (MRS) [Copestake et al., 2005]
- Constraint Language for Lambda Structures (CLLS) [Egg et al., 2001]
- Dominance Constraints [Althaus et al., 2003]
- Hole Semantics [Bos, 1995]
- Underspecification in LTAG Semantics [Kallmeyer and Romero, 2008]
- Logical Description Grammar [Muskens, 2001]

Describing Ambiguity (7)

Now consider

(7) Every researcher of a company saw most samples

(8)



Challenge: impossible scope orders should be correctly excluded by the underspecified representations. (8) correctly excludes $\forall > most' > \exists$.

Deriving Ambiguity (1)

Deriving ambiguity

Approach:

- First, give an initial description of the readings containing scope-bearing expressions with a not yet determined scope.

  (9) Every woman loves a man
  $love'(\langle forall'x.woman'(x)\rangle, \langle exists'y.man'(y)\rangle)$

- To derive a set of fully specified representations from such a description, a resolution algorithm integrates terms into descriptions by discharging them.

Deriving Ambiguity (2)

For $\forall > \exists$, the existential term is integrated first: the term is replaced by the bound variable and the quantifier with the term's bound variable and restriction is prefixed to the resulting expression.

(10) $exists'y[man'(y), love'(\langle forall'x.woman'(x)\rangle, y)]$

Integrating the universal term then yields $\forall > \exists$.

[Hobbs and Shieber, 1987] present an algorithm for more complicated cases, in particular, nested quantification.

The difference between underspecification formalisms that describe the readings of an ambiguous expression and those that derive these readings is that in the latter the algorithm is essential in determining the set of solutions.

Processing Underspecified Representations (1)

Underspecified semantic representations can be further processed in order to derive fully specified (or at least less ambiguous) semantic representations.

- One can enumerate the readings by resolving the constraints with the help of so-called solvers. Such solvers are available, e.g. for MRS representations and for the language of dominance constraints [Koller et al., 1998].
- Related to this is work on redundancy elimination, which weeds out spurious ambiguities either during the resolution process or directly on the underspecified representations.
- Some underspecified semantic representations allow the deduction of fully specified information. For example, if Amélie is a woman, then it follows from (1) that she loves a man, no matter which reading of (1) is at stake.

Processing Underspecified Representations (2)

Specific readings can be chosen (or the number of potential readings be reduced) if one strengthens underspecified representations by preferences for specific kinds of readings. Types of preferences:

- Syntactic preferences: surface linear order, c-command.
- Preferences based on grammatical functions and thematic roles. For example, a scope preference hierarchy stretching from topic (strongest preference for wide scope) over subject and PP complement down to object.
- The determiners themselves also have different tendencies to take wide scope: the hierarchy ranges from *each* and *every* (strongest preference for wide scope) down to *a few*.

References

[Althaus et al., 2003] Althaus, E., Duchier, D., Koller, A., Mehlhorn, K., Niehren, J., and Thiel, S. (2003). An efficient graph algorithm for dominance constraints. *Journal of Algorithms*, 48(1):194–219.

[Bos, 1995] Bos, J. (1995). Predicate logic unplugged. In Dekker, P. and Stokhof, M., editors, *Proceedings of the 10th Amsterdam Colloquium*, pages 133–142.

[Copestake et al., 2005] Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. (2005). Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3:281–332.

[Egg, 2010] Egg, M. (2010). Semantic underspecification. *Language and Linguistics Compass*, 4(3):166–181.

[Egg et al., 2001] Egg, M., Koller, A., and Niehren, J. (2001). The constraint language for lambda-structures. *Journal of Logic, Language, and Information*, 10:457–85.

[Hobbs and Shieber, 1987] Hobbs, J. R. and Shieber, S. M. (1987). An algorithm for generating quantifier scopings. *Computational Linguistics*, 13:47–63.

[Joshi et al., 2003] Joshi, A. K., Kallmeyer, L., and Romero, M. (2003). Flexible composition in LTAG: Quantifier scope and inverse linking. In Bunt, H., van der Sluis, I., and Morante, R., editors, *Proceedings of the Fifth International Workshop on Computational Semantics IWCS-5*, pages 179–194, Tilburg.

[Kallmeyer and Romero, 2008] Kallmeyer, L. and Romero, M. (2008). Scope and situation binding in LTAG using semantic unification. *Research on Language and Computation*, 6(1):3–52.

[Koller et al., 1998] Koller, A., Niehren, J., and Treinen, R. (1998). Dominance Constraints: Algorithms and Complexity. In *Proceedings of the Third International Conference on Logical Aspects of Computational Linguistics (LACL '98)*, Grenoble, France.

[Muskens, 2001] Muskens, R. (2001). Talking about trees and truth-conditions. *Journal of Logic, Language, and Information*, 10:417–55.

[Reyle, 1993] Reyle, U. (1993). Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics*, 10:123–179.

[Reyle, 1996] Reyle, U. (1996). Co-Indexing Labelled DRSs to Represent and Reason with Ambiguities. In van Deemter, K. and Peters, S., editors, *Semantic Ambiguitiy and Underspecification*, number 55 in CSLI Lecture Notes, pages 239–268. CSLI, Stanford.