20

Semantic Morphology

BJÖRN GAMBÄCK

Semantic Morphology addresses the problem of designing the rules needed for mapping between the semantic lexicon and semantic grammar. The text discusses the relation between semantics, lexicon, and morphology in unification-based grammars and builds on the current trends in Computational Semantics to use underspecification and compositionality. The approach to Semantic Morphology advocated here assumes compositional word formation from (semantic) word roots and affixes that are given their own entries in the semantic lexicon. Different feature usages are then utilized to reach the intended surface word-form matches, with the correct feature settings.

20.1 Introduction

The interaction between morphology and the (syntactic) lexicon on one side and the (syntactic) grammar on the other has been discussed at length in various papers and for various languages. However, the parentheses in the previous sentence point to an almost general restriction: the treatment of language structure has focused mainly on the problems relating morphology to syntax, while little attention has been given to the semantics.

With Semantic Morphology we do not mean the issue of how the actual word-forms are located in the input string, but will take for granted that a module is available to do this work in a unification-based grammar setting, for example such a "lazy" implementation of two-level style morphology (Koskenniemi, 1983) as the one of Carter (1995). Thus in essence, there should be a separation of the task of identifying the input word-form and the task of mapping the lexical feature settings into the grammar, as also argued by Trost and Matiasek (1994).

Inquiries into Words, Constraints and Contexts.
Antti Arppe et al. (Eds.)

The rest of the text will address the issues of designing and implementing unification-based semantic morphological processing. That is, the morphological rules that execute the mapping between the semantic lexicon and (the rest of!) the semantic grammar — and the way in which features can be used in order to restrict the output to only the desired forms. In doing so, some practical implementations will be discussed, in particular for Japanese and Swedish. Firstly, though, we should note that there have been three strong trends in the Computational Linguistic community during the last decades, both in unification-based grammar approaches in general as well as in most approaches to Computational Semantics:

1. keep as much as possible of the semantic information lexicalized,
2. build complex structures in a compositional manner, and
3. postpone decisions as long as possible.[1]

The first two trends are the topics of the next section, while the third trend is discussed in Section 20.3. Then Section 20.4 introduces some of the work on separating out Semantic Morphology, while Sections 20.5 and 20.6 go into some examples for Japanese and Swedish, respectively. Finally, Section 20.7 sums up the discussion.

20.2 Lexicalization and compositionality

The trend to keep most of the information in the lexicon (rather than in the grammar rules, as traditionally) aims to keep the grammar rules as simple as possible and the number of distinct grammar rules as low as possible — which in turn may result in rather complicated lexica; lexica that are hard, or even impossible, to clearly separate from the grammar proper. On the morphology side, the solution adopted here is the one of introducing affixes as lexical categories, that is, that word formation is given as a compositional addition of affixes to the word roots.

Compositionality may be defined rather strictly so that the interpretation of a phrase always should be the (logical) sum of the interpretations of its subphrases. A semantic formalism being compositional in this strict sense would also trivially be monotonic, since no destructive changes would need to be undertaken while building the interpretation of a phrase from those of its subphrases.[2] In effect then, all the information from the terminal nodes would be passed up to the input (top-level) nodes of the grammar.

[1] A fourth strong trend has been to do away with all "deep" level processing and only use shallow rules or statistical models. However, a discussion of the treatment of morphology in such a "shallow" approach is outside of the scope of this text.

[2] A semantic representation is monotonic if and only if the interpretation of a category on the right side of a rule subsumes the interpretation of the left side of the rule.

However, compositionality is more commonly defined in a wider sense, allowing for other mappings from subphrase-to-phrase interpretation than the sum, as long as the mappings are such that the interpretation of the phrase still is a function of the interpretations of the subphrases. A common such mapping is to let the interpretation of the phrase be the interpretation of its (semantic) head modified by the interpretations of the adjuncts. If this modification is done by proper unification, the monotonicity of the formalism will still be guaranteed.

In general we need morphology and grammar rules for addition of already manifest semantic information (e.g., from the lexicon) and ways of passing non-manifest information (e.g., about complements sought). Assuming a normalised structure, we can then allow for information passing in three ways: trivial composition, function-argument application, and modifier-argument application. The trivial composition manifests itself mainly in rules that are inherently (semantically) unary branching. That is, rules that either are syntactically unary branching, or where the semantics of at most one of the daughter (right-hand side) nodes need to influence the interpretation of the mother (left-hand side) node.

The two types of application rules are quite similar to each other and appear on all (semantically) binary branching rules of the grammar. In both application rule types, the bulk of the semantic information is passed to the mother node from the semantic head among the daughter nodes. However, in functor-argument application the functor is the semantic head, while in modifier-argument application the argument is the semantic head.

The main difference between the two types pertains to the (semantic) subcategorisation schemes: In functor-argument application, the functor subcategorises for the argument, the argument may optionally subcategorise for the functor, and the mother's subcategorisation list is the same as the functor's, minus the argument. Letting **main** intuitively identify the semantic information, **subcat** the subcategorisation list, and *Functor* the semantic head, we get:

(1)

Mother		*Functor*		*Argument*
main	$\boxed{1}$	**main**	$\boxed{1}$	**main** $\boxed{2}$
subcat	$\langle \boxed{3} \rangle$	**subcat**	$\langle \boxed{2} \mid \boxed{3} \rangle$	**subcat** $\langle \boxed{1} \rangle$

$\Rightarrow$

In modifier-argument application, the modifier subcategorises for the argument (only), while the argument does not subcategorise for the modifier; its subcategorisation list is passed unchanged to the mother node. This is shown schematically in (2), with *Argument* being the semantic head:

(2)

Mother		*Modifier*		*Argument*
main	$\boxed{1}$	**main**		**main** $\boxed{1}$
subcat	$\langle \boxed{2} \rangle$	**subcat**	$\langle \overline{\boxed{1}} \rangle$	**subcat** $\langle \boxed{2} \rangle$

$\Rightarrow$

20.3 Ambiguity and underspecification

The third trend concerning postponing decisions relates to the problem of ambiguity. Amongst others, ambiguity in a natural language expression may be due to the fact that one of the words used may not have a unique meaning, that more than one syntactic structure may be assigned to the expression, or that the scope relations are not clear. Ambiguities of this kind decrease processing efficiency, since usually all of the possible interpretations have to be assumed to be right until hard facts prove the contrary. The bad news is that this normally happens after a lot of processing has been done.

A way around this dilemma is to have a common representation for all of the possible interpretations of an ambiguous expression, as in the so-called Quasi-Logical Form notation introduced by Alshawi and van Eijck (1989). Following Reyle (1993), the term *underspecification* has been the accepted one to describe this idea. The basic strategy is not to use representations that encode a concrete interpretation but a *set* of interpretations. Thus, the representations are underspecified with respect to one single specific interpretation.

Most work on underspecification has concentrated on scopal ambiguities and anaphora; however, Pinkal (1996) extends the theory of underspecification and discusses several phenomena that lend themselves to this type of compact representation: local ambiguities (e.g., lexical ambiguities, anaphoric or deictic use of pronouns), global ambiguities (e.g., scopal ambiguities, collective-distributive readings), and ambiguous or incoherent non-semantic information (e.g., PP-attachment, number disagreement). Another argument (in addition to the issues related to processing) for underspecified representations is the observation that there is evidence that humans use underspecified information when dealing with natural language. Pinkal (1999) gives a good overview of different approaches to underspecification and also argues at length for its cognitive motivations based on the fact that humans are able to draw inferences from underspecified semantic information.

In order to represent underspecifation, we will assume a semantic representation language such as the ones described by Bos et al. (1996) and Copestake et al. (1999), that is, a language of 'flat' structures which assigns a unique label (name) to every basic formula of the object language with scope (appearing on quantifiers and operators) being represented in an underspecified way by variables ranging over labels. The labeling of conditions is used to make it easier to refer to a particular condition, enabling us to state constraints on the relations between the conditions at the meta-level.

For building these representations we use the operations described above in order to compositionally combine simple representations into complex ones. In addition, a we use a three-place structure referred to as the *context*. It contains the representation's main instance, **inst** (the label of the main event,

which normally is the verb) and two functions that help us keep track of a couple of special labels. These are **main**, the label of the semantic head of the representation, and **top**, the top-most label of the semantic structure.

20.4 Related work

One reason for the lack of interest in computational semantic morphology is that there is a straightforward way to completely ignore it! A common solution is to let the syntactic part of the morphology do all the work and let the semantics "piggyback" on that, letting the semantic lexicon handle the cases where this cannot be done. Accordingly, the German version of the Verbmobil grammar (Bos et al., 1996) let the syntax resolve all inflectional affixing, while verb prefixing (which is rich in German) was fully specified in the lexicon. This means that, e.g., *durchlaufen* (run through) and *durchleben* (live through) need two separate entries in the semantic lexicon, neither of which relate directly to the compositional parts. Thus the "straightforward" solution is possible, but neither elegant nor implementationally attractive. It makes more sense to allow each of the different parts of the word to have their own entries in the semantic lexicon and to apply semantic morphological rules to the parts in order to build the overall semantic interpretation of the word.

There has been some work on relating morphology to semantics within the Lexical Functional Grammar (LFG) and Head-Driven Phrase Structure Grammar (HPSG) traditions. In LFG, Sadler and Nordlinger (2006) argue for treating the problem of case-stacking[3] by connecting the morphology to LFG's functional descriptions in a tree-based fashion. Andrews (2005) argues against this and instead proposes a flat notation. In the HPSG school, most work on semantics has during the last decade concentrated on (flat) Minimal Recursion Semantics, MRS (Copestake et al., 1999). However, these efforts have mainly been devoted to the grammar as such and have more or less disregarded the morphological semantics. The main exceptions to this concern the work on HPSG for Japanese (e.g. Siegel and Bender, 2002).

A recent alternative to MRS is LRS, Lexical Resource Semantics (Sailer, 2004) which aims to separate out the description of local semantic phenomena (such as selectional restrictions and linking, the mapping between semantic roles and syntactic complements) from the non-local (clausal) semantics. In effect, the representation of local semantics in LRS takes the "semantic-head based resolution" of Gambäck and Bos (1998) as a starting point, but extends it and formalises it. Riehemann (1998) argues for an approach in which generalisations from existing words are expressed as schemata, organised in an HPSG-style inheritance network. This is attractive and elegant, although efficiency of an implementation of it still has to be demonstrated.

[3] When a single word contains multiple case markers.

20.5 Japanese morphology

Here we will instead adopt an alternative solution to morphology, where affixes are given specific lexical entries. A very clear example of this kind of treatment can be seen in Japanese. Japanese verbs exhibit affix-based inflectional morphology in its own right, but also more specific phenomena such as the usage of light verbs and particles (especially postpositional) are common. By including the verbal affixes in the semantic lexicon we can treat them and the postpositional particles in a uniform way. Consider as an example the verb phrase *haitte orimasu* in (3).

(3) *itsumo iroiro kaigi ga hait- te ori- masu*
always various meeting NOM be-put-in PART ASP HON+PRES
'all types of meetings are scheduled every day'

Here *hait* is the main verb and *ori* an auxiliary, while *te* and *masu* are inflectional affixes. The core semantical information comes from the main verb, so that the affixes can be treated as modifiers of the respective verb and the auxiliary as a modifier of the main verb. Thus we can, for example, let the lexical entry for *masu* mainly introduce the semantic information for representing the honorific form and pass it up in a purely compositional manner in the morphological analysis tree. The lexical entry for the honorific affix would basically look as (4). So, the only argument which *masu* subcategorises for is its verb, which in turn introduces the discourse marker labeled as [1].

(4)
$$\left[\begin{array}{ll}
\text{RELN} & \texttt{masu} \\
\textit{context} & \left[\begin{array}{ll} \textbf{inst} & \boxed{1} \\ \textbf{main} & \boxed{2} \\ \textbf{top} & \boxed{3} \end{array}\right] \\
\text{TENSE-ASPECT} & \left[\begin{array}{ll} \textbf{main} & \boxed{2} \\ \text{TENSE} & \text{present} \\ \text{HONORIFIC} & \texttt{masu} \end{array}\right] \\
\text{MODIFIER} & \text{yes} \\
\text{SUBCAT} & \left\langle \left[\textit{context} \left[\begin{array}{ll} \textbf{inst} & \boxed{1} \\ \textbf{main} & \\ \textbf{top} & \boxed{3} \end{array}\right] \right] \right\rangle
\end{array}\right]$$

The most important part of the entry in (4) is the feature-bundle designated TENSE-ASPECT. Here it introduces two things, the present tense and the honorific level which can be viewed as a sort of aspectual information. The honorific level is set simply to `masu` and will in due time be bound to the discourse marker by the **main** label of *masu*, [2] — thus the lexical entry in effect

introduces a honorific aspect on the main verb. There is no need to refer to the **main** label of the main verb (shown by the '__'), but its **top** label [3] is bound to the **top** label of *masu*, meaning that the honorific aspect and the present tense will take the same scope as the verb (i.e., normally over the entire sentence). This is not very important for the present discussion, but obviously not a necessary restriction. Bonami (2001) suggests including an (underspecified) scopal restriction in the lexical entry for the tense relation itself, allowing it to take a different scope than the other elements of the tense-aspect structure.

In the same fashion, *ori* would introduce a progressive aspect, while the affix *te* basically would not add anything to the semantics. The verb *hait* is in itself intransitive and thus subcategorises for one argument, the subject. The entire verb phrase structure would then be built recursively using the modifier application rule of Section 20.2. Filling in the schematic rule (2) on Page 206 gives us an overall structure like the one in (5).

(5)
$$\left[\begin{array}{ll}
\text{RELN} & \texttt{hait} \\
\textit{context} & \left[\begin{array}{ll} \textbf{inst} & [1] \\ \textbf{main} & [2] \\ \textbf{top} & [3] \end{array}\right] \\
\text{TENSE-ASPECT} & \left[\begin{array}{ll} \textbf{main} & [2] \\ \text{TENSE} & \text{present} \\ \text{ASPECT} & \text{progressive} \\ \text{HONORIFIC} & \texttt{masu} \end{array}\right] \\
\text{MODIFIER} & \text{no} \\
\text{SUBCAT} & \left\langle \left[\textit{context} \left[\begin{array}{ll} \textbf{inst} & [4] \\ \textbf{main} & _ \\ \textbf{top} & [3] \end{array}\right] \right] \right\rangle \\
\text{ROLE} & \texttt{subject(}[2]\texttt{,}[4]\texttt{)}
\end{array}\right]$$

Nicely enough, we would need to make no principal distinction between the applications of the affixes to the verbs and the application of the auxiliary to the main verb. Quite importantly, there would also be no fundamental distinction between the behaviour of these morphology level rules and the rules, for example, for the application of postpositions to NPs to build PPs.

The basic construction in the Japanese syntax is the PP. A PP may be constructed in a range of different ways, the base case, however, being PP $\rightarrow$ NP P. Semantically, the P in this rule is treated uniformly (for all types of postpositions) as a functor applying to the NP, that is, using the functor-argument application rule (1) shown schematically on Page 206.

20.6 Swedish morphology

For an inflectional language like Swedish, where, for example, most of the tense and aspect information can be found in the suffix of the main verb, it is natural to view the tense-aspect information as forming a function of the affix — the information is then filtered up from the verbal affix to the verb phrase. Most work on morphology for Swedish and other Scandinavian languages has concentrated on the purely syntactic side (e.g. Karlsson, 1992; Gambäck, 2001). However, the treatment of non-compositional Danish phrasal verbs in PAROLE/SIMPLE by Pedersen and Nimb (2000) follows the same lines as here by advocating a "split late" strategy where phrasal verbs are singled out as late as possible in the morphological processing, that is, in the semantic part of it.

The lexicon form of choice for Swedish verbs is the imperative, since this form constitutes the stem of most other inflections. For tense and aspect purposes, however, the imperative is a bit peculiar: it stands almost on the side of the entire tense-aspect system. Thus the lexicon contains stems for which the tense-aspect information is only partially instantiated. The (normally) full instantiation is obtained by the inflection in morphology rules as the following schematic one:

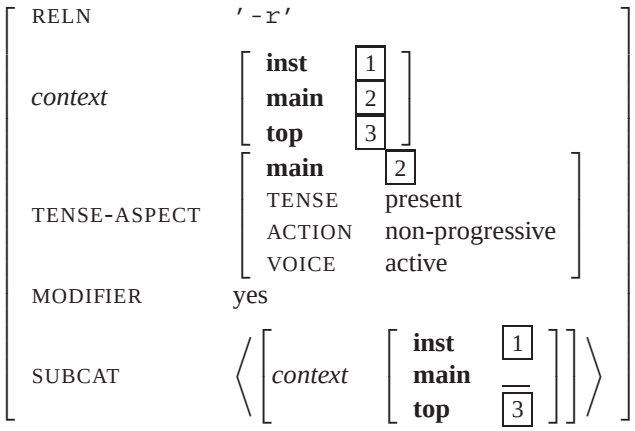
(6)

Mother		*Verb*		*Suffix*	
main	$\boxed{1}$	**main**	$\boxed{1}$	**main**	$\overline{}$
ten-asp	$\boxed{3}$	**ten-asp**	$\overline{}$	**ten-asp**	$\boxed{3}$
subcat	$\langle \boxed{2} \rangle$	**subcat**	$\langle \boxed{2} \rangle$	**subcat**	$\langle \boxed{1} \rangle$

$\Rightarrow$

where the mother verb is formed by adding a suffix to the daughter verb (i.e., the stem form). The tense-aspect information from the suffix is passed up to the inflected verb. This is also the only (semantic) information added by the suffix; the other parts of the mother-verb semantics come from the daughter. An example of a suffix entry is the one in (7) for the ending '-r', which is used to form the present tense when added to the stem of verbs belonging to the first (e.g., *menar*) and third declension (*sker*) as well as those belonging to the third subgroup of the fourth declension (*ser*).

Just like the rules for affixing, we can allow for rules, for example, for the construction of particle verbs simply by including the particle on the (semantic) subcategorisation list of the verb and having a semantic morphology rule for V $\rightarrow$ P V. Wolters (1997) thus proposes a solution to the German prefix verb problem (Section 20.4) in which each verb's lexical entry contains an indication of which prefixes it may combine with in an HPSG framework. Or rather, which *senses* of the prefixes a verb may combine with in order to form specific interpretations.

(7)


20.7 Summary

The text has advocated singling out Semantic Morphology as a topic in its own right. This contrasts with many approaches to unification-based grammars where syntax and semantics are treated in parallel, as well as with approaches where the syntax takes total control of the morphology. A key aspect of the treatment presented here is to introduce affixes as their own entries in the semantic lexicon.

Acknowledgements

This work was influenced by the views and efforts of many people: several of my former colleagues and most of the people mentioned in the reference list. Thanks to all of you.

References

Alshawi, Hiyan and Jan van Eijck. 1989. Logical forms in the Core Language Engine. In *Proc. 27th Annual Meeting of the Association for Computational Linguistics*, pages 25–32. ACL, Vancouver, British Columbia.

Andrews, Avery D. 2005. F-structural spellout in LFG morphology. Manuscript. Australian National University, Canberra, Australia.

Bonami, Olivier. 2001. A syntax-semantics interface for tense and aspect in French. In F. van Eynde, L. Hellan, and D. Beermann, eds., *Proc. 8th International Conference on Head-Driven Phrase Structure Grammar*, pages 31–50. Trondheim, Norway: CSLI Publications.

Bos, Johan, Björn Gambäck, Christian Lieske, Yoshiki Mori, Manfred Pinkal, and Karsten Worm. 1996. Compositional semantics in Verbmobil. In *Proc. 16th International Conference on Computational Linguistics*, vol. 1, pages 131–136. ACL, København, Denmark.

Carter, David. 1995. Rapid development of morphological descriptions for full language processing systems. In *Proc. 7th Conference of the European Chapter of the Association for Computational Linguistics*, pages 202–209. ACL, Dublin, Ireland.

Copestake, Ann, Dan Flickinger, Ivan Sag, and Carl Pollard. 1999. Minimal Recursion Semantics: An introduction. Manuscript. CSLI, Stanford, California.

Gambäck, Björn. 2001. Unification-based lexicon and morphology with speculative feature signalling. In A. Gelbukh, ed., *Computational Linguistics and Intelligent Text Processing: Proc. 2nd International Conference*, no. 2004 in Lecture Notes in Computer Science, pages 349–362. Mexico City, Mexico: Springer-Verlag.

Gambäck, Björn and Johan Bos. 1998. Semantic-head based resolution of scopal ambiguities. In *Proc. 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pages 433–437. ACL, Montreal, Canada.

Karlsson, Fred. 1992. SWETWOL: A comprehensive morphological analyser for Swedish. *Nordic Journal of Linguistics* 15(1):1–45.

Koskenniemi, Kimmo. 1983. *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*. Doctor of Philosophy Thesis, University of Helsinki, Dept. of General Linguistics, Helsinki, Finland.

Pedersen, Bolette Sandford and Sanni Nimb. 2000. Semantic encoding of Danish verbs in SIMPLE: Adapting a verb-framed model to a satellite-framed language. In *Proc. 2nd International Conference on Language Resources and Evaluation*, pages 1405–1412. ELRA, Athens, Greece.

Pinkal, Manfred. 1996. Radical underspecification. In P. Dekker and M. Stokhof, eds., *Proc. 10th Amsterdam Colloquium*, vol. 3, pages 587–606. Amsterdam, Holland.

Pinkal, Manfred. 1999. On semantic underspecification. In H. Bunt and E. Thijsse, eds., *Proc. 3rd International Workshop on Computational Semantics*, pages 33–56. Tilburg, Holland.

Reyle, Uwe. 1993. Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics* 10:123–179.

Riehemann, Susanne. 1998. Type-based derivational morphology. *Journal of Comparative Germanic Linguistics* 2:49–77.

Sadler, Louisa and Rachel Nordlinger. 2006. Case stacking in realizational morhology. *Linguistics* (to appear).

Sailer, Manfred. 2004. Local semantics in Head-Driven Phrase Structure Grammar. In O. Bonami and P. C. Hofherr, eds., *Empirical Issues in Formal Syntax and Semantics*, vol. 5, pages 197–214. Paris, France: CNRS.

Siegel, Melanie and Emily M. Bender. 2002. Efficient deep processing of Japanese. In *Proc. 3rd Workshop on Asian Language Resources and International Standardization*, pages 31–38. ACL, Taipei, Taiwan.

Trost, Harald and Johannes Matiasek. 1994. Morphology with a null-interface. In *Proc. 15th International Conference on Computational Linguistics*, vol. 1, pages 141–147. ACL, Kyoto, Japan.

Wolters, Maria. 1997. Compositional semantics of German prefix verbs. In *Proc. 35th Annual Meeting and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 525–527. ACL, Madrid, Spain. Student session.