

## Abstract

In this paper I will make an attempt to show how the linking normally done in the lexicon also can be done in constructions. The motivation behind this is the flexibility it gives the grammar writer in underspecifying lexical entries. Being too rigid about linking in the lexicon may lead to unsatisfying results such as multiple lexical entries for what one intuitively feels is just one lexical entry, or alternatively, lexical rules which are not morphologically motivated. The aim is to show that this can be avoided by letting constructions introduce the linking information instead.

## 1 Introduction

First I will briefly present what I mean by linking in the lexicon. Second I will discuss the kind of semantic representations I will use. Third I will discuss valence features and show how constructions adapted to the function of the argument may be used to introduce linking information. Then I will discuss two problems that turn up when one tries to do linking in the constructions, namely long distance dependencies and linking of subjects.

The fundament for the work presented here is the Matrix Grammar (Bender et al. [2002]) (Version 0.6), which is a subset of the English Resource Grammar (ERG) and which purpose is to give a grammar writer the most basic types needed to write a large coverage HPSG grammar using the LKB system (Copestake [2001]). The Matrix has build into it MRS semantics (Copestake et al. [1999]).

## 2 Linking in the lexicon

In an active sentence with a ditransitive verb like *give*, there is linking between the ARG1 role of the verb relation and the index of the subject. In addition the index of the direct object is linked to the ARG2 of the verb relation and the index of the indirect object is linked to the ARG3 of the verb relation. Normally this linking is done in the lexicon, and the linking information of a ditransitive lexeme like *give* will look like in figure (1)<sup>1</sup>.

One problem about doing linking in the lexicon is that one at an early stage has to decide how many semantic arguments the verb has. In the case of *give*, one will need another lexical entry if one wants to say that that it has only two semantic arguments. Another example is the Norwegian the verb *kaste* (throw) which may be intransitive, transitive and ditransitive. It may also have a particle or a PP as complement. If one wants to decide in the lexicon about the semantic arguments of this verb, one is forced to assume many lexical entries (or lexical rules).

---

<sup>1</sup>The feature KEYREL is a pointer to the main relation in a lexeme.

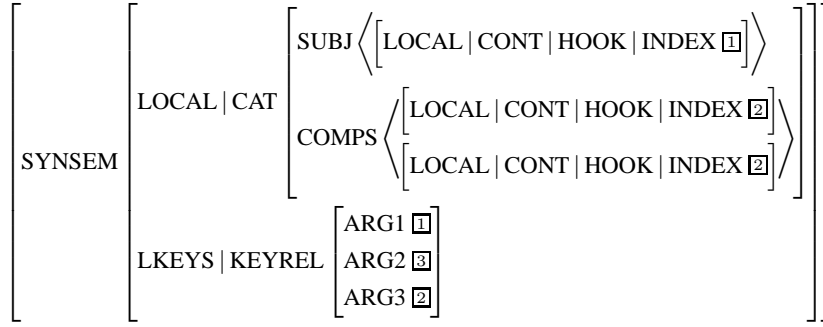


Figure 1: Linking type for ditransitive verbs

### 3 Decomposed semantics

Instead of doing linking in the lexicon, I would suggest to underspecify lexical entries with regard to linking, and rather do the linking in the constructions. In the following I will use a decomposed semantics called RMRS (Robust Minimal Recursion Semantics) (Copestake [2003]). Argument taking lexemes have a basic relation with a handle (LBL), an index (ARG0) and a PRED feature. There will be other basic relations that link the main relation to its arguments. These relations will have a handle and a feature ARG1, ARG2, ARG3 or ARG4. The semantic representation of the sentence *a man admires a painting* looks like in figure (2)<sup>2</sup>.

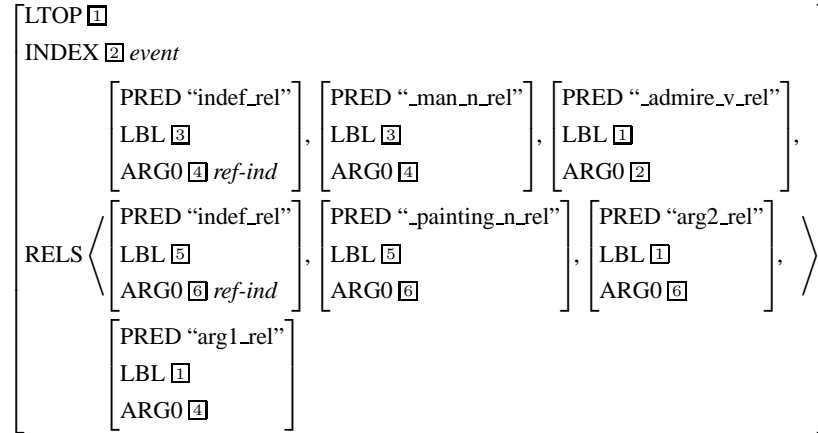


Figure 2: RMRS for “A man admires a painting”

Here the relation of the transitive verb, `_admire_v_rel`, shares handle with the

<sup>2</sup>Quantifier scope will not be represented in this paper

arg1\_rel and the arg2\_rel. The representation above is equivalent to the MRS representation in figure (3):

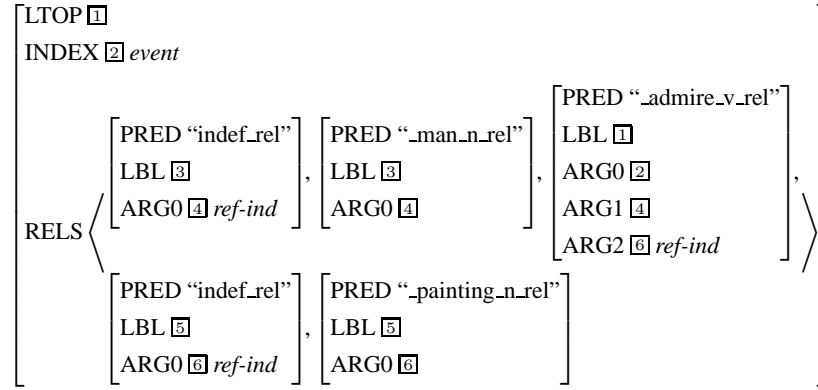


Figure 3: MRS for “A man admires a painting”

## 4 Valence constructions

In order to make the valence rules introduce linking relations, I will have to make some assumptions about the valence lists. The type *valence* in the Matrix has the following definition:

```
valence := valence-min &
[ SUBJ list,
  SPR list,
  COMPS list,
  SPEC list,
  --KEYCOMP avm ].
```

In this approach the type *valence* is changed:

```
valence := valence-min &
[ SPR list,
  DOBJ list,
  IOBJ list,
  POBJ list ].
```

The two lists SUBJ and SPR are merged into SPR. The COMPS list is divided, so that there is one list for each kind of complement. There is one list for direct objects, one for indirect objects, and one for other complements such as particles and PPs. The SPEC list is taken out.

The decomposed semantics together with the new valence lists allow me to write valence rules that do linking between argument taking signs and their arguments. The rule for direct objects is given in (4)<sup>3</sup>:

<sup>3</sup>The exclamation marks on the RELS list mean that the list is a difference list

$$\left[ \begin{array}{l}
\text{head-dobj-phrase} \\
\text{SYNSEM} \mid \text{LOCAL} \left[ \begin{array}{l}
\text{CAT} \left[ \begin{array}{l}
\text{HEAD } \boxed{0} \text{ prep-or-verb} \\
\text{VAL} \mid \text{DOBJ} \langle \rangle
\end{array} \right] \\
\text{CONT} \mid \text{HOOK } \boxed{1} \left[ \text{LTOP } \boxed{2} \right]
\end{array} \right] \\
\text{ARGS} \left\langle \boxed{3} \mid \text{S} \mid \text{L} \left[ \begin{array}{l}
\text{CAT} \left[ \begin{array}{l}
\text{HEAD } \boxed{0} \\
\text{VAL} \mid \text{DOBJ} \langle \boxed{4} \rangle
\end{array} \right]
\end{array} \right] \right\rangle, \left[ \text{S } \boxed{4} \mid \text{L} \mid \text{CONT} \mid \text{HOOK} \mid \text{INDEX } \boxed{5} \right] \rangle \\
\text{HEAD-DTR } \boxed{3} \\
\text{C-CONT} \mid \text{RELS} \left\langle \begin{array}{l}
\text{arg2-relation} \\
\text{LBL } \boxed{2} \\
\text{PRED "arg2\_rel"} \\
\text{ARG2 } \boxed{5}
\end{array} \right\rangle
\end{array} \right]$$

Figure 4: Head direct object rule

The feature C-CONT in a phrase has as value the type *mrs* and it allows the grammar writer to let constructions introduce relations in the same way as lexical entries do. In the phrase above, a linking relation (arg2-relation) is introduced in C-CONT. The handle of the linking relation is linked to the LTOP of the phrase. This ensures that it has the same handle as the main relation of the head. The ARG2 of the linking relation is linked to the index of the direct object. This means that the index of the direct object will be the ARG2 of the main relation of the head.

## 5 Long distance dependencies

One possible problem connected to letting constructions introduce linking relations is long distance dependencies. In long distance dependencies the filler phrase will not have access to information about the function of the filler, and one cannot predict which linking relation it should have. However, this is not a problem if one lets the extraction rule do the linking. In figure (5) I suggest a hierarchy of direct object phrases, where the most essential information is captured in a supertype of the *head-dobj-phrase* and the *extr-dobj-phrase*.

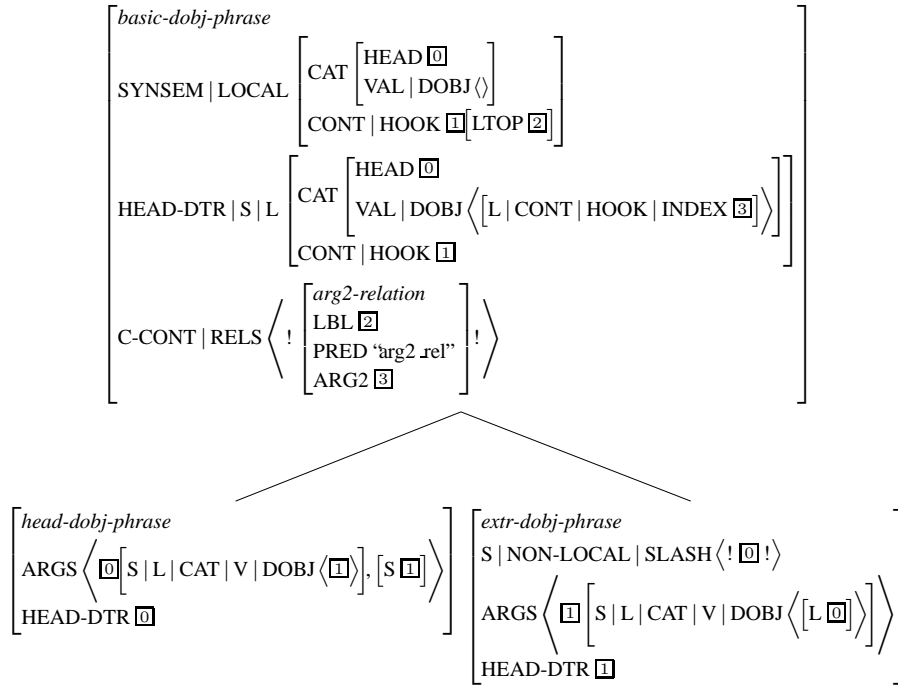


Figure 5: Type hierarchy of direct object phrases

## 6 Linking of subjects

In this section I will present some problems that subjects in Norwegian raise.

### 6.1 Presentational constructions

Presentational constructions are very frequent in Norwegian. In presentational constructions the expletive *det* functions as subject. This word is not assumed to have any content, the content that the presentational construction may contribute, can be ascribed the construction, and not the word. Since the expletives don't have any content, there is no need for a linking relation. So I have to assume two kinds of subject rules, one which introduces a linking relation, and one which doesn't. In the first case the subject has a *ref-ind* as an index, and in the second case it will have an *expl-ind*. Then it is an expletive.

### 6.2 Subjects with content

It is not always the case that subjects with an referential index are in an ARG1 relation to the verb. They can also be in an ARG2 or ARG3 relation to the verb. This

may be the case when the verb has passive voice or it is an ergative. The Norwegian examples (1)-(3) illustrate this fact. The sentences also have presentational counterparts (4) and (5), where the linking should be unproblematic.

- (1) *En mann kommer*  
a man comes  
'A man comes'
- (2) *Mannen blir gitt en kake.*  
man-the becomes given a cake  
'The man is given a cake.'
- (3) *En kake blir gitt mannen.*  
a cake becomes given man-the  
'The man is given a cake.'
- (4) *Det kommer en mann.*  
it comes a man  
'A man comes.'
- (5) *Det blir gitt mannen en kake.*  
it becomes given man-the a cake  
'The man is given a cake.'

In order to account for the different linking relations that can be introduced by the subject rule, I introduce a feature **VOICE** on *head*. The value of the **VOICE** feature, *voice* has the feature **SPR-CONT**. The value of **SPR-CONT** is *mrs*. The type *head* now has the following information:

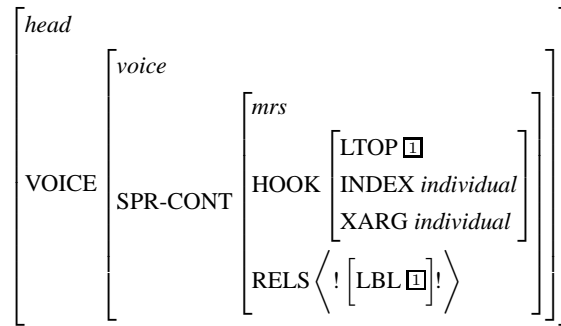


Figure 6: The **VOICE** feature on the type *head*

The head specifier rule for non-expletive subjects (*content-spr-phrase*) is similar to the direct object rule. Analogous with the direct object phrases, there will be a general specifier phrase that holds for both the *head-spr-phrase* and the *extr-spr-phrase* (7).

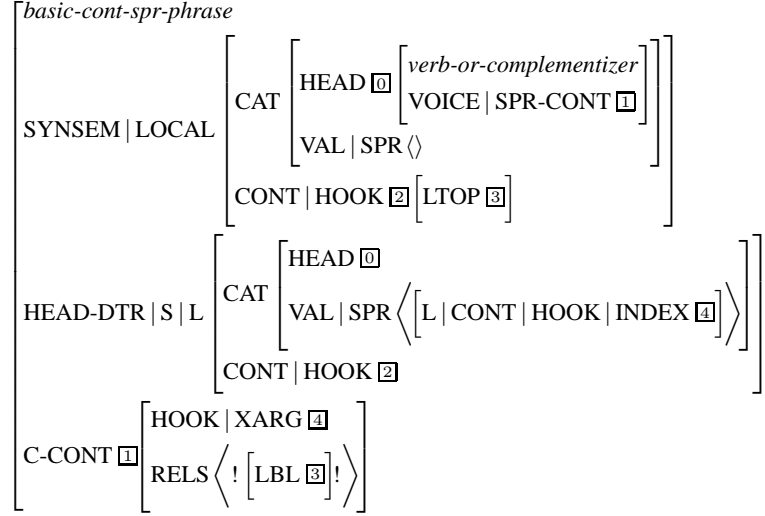


Figure 7: Basic content specifier phrase

The difference is that in this phrase it is underspecified which linking relation is introduced by C-CONT. Instead the C-CONT of the construction is unified with the SPR-CONT of the head projection. Given the hierarchy in (8) and the type definition in (9) we can infer that a sentence with an active agentive verb will get an arg1-relation in the C-CONT of the subject rule.

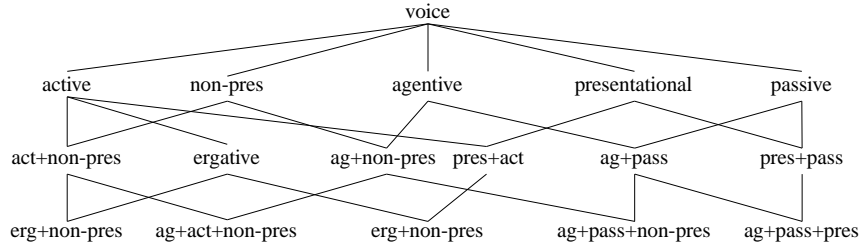


Figure 8: Hierarchy under *voice*

I can also use the new machinery to account for the linking relations for passives and ergatives that have a subject which not an expletive. This can be done by assuming a unary rule that moves one of the objects into the specifier list, constrains the VOICE value to be *non-pres* and specifies the relation in SPR-CONT to be an arg2-relation in case of direct objects, and an arg3-relation in case of indirect objects.

$$\left[ \begin{array}{c} \text{ag+act+non-pres} \\ \text{SPR-CONT} \left[ \begin{array}{c} \text{HOOK} \mid \text{XARG } \boxed{\phantom{x}} \\ \text{RELS} \left\langle ! \left[ \text{ARG1 } \boxed{\phantom{x}} \right] ! \right\rangle \end{array} \right] \end{array} \right]$$

Figure 9: Type definition of *ag+act+non-pres*

## 7 Conclusion

I have shown that it is possible to underspecify lexical entries with regard to linking information and still account for phenomena like discontinuous constituents and subjects of ergative and passive verbs.

## References

- Emily M. Bender, Daniel P. Flickinger, and Stephan Oepen. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei, Taiwan, 2002.
- Ann Copestake. *Implementing Typed Feature Structure Grammars*. CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, 2001. URL <http://cslipublications.stanford.edu/site/1575862603.html>.
- Ann Copestake. Report on the design of rmrs (preliminary version). Technical report, Cambridge, April 11, 2003.
- Ann Copestake, Daniel P. Flickinger, and Ivan A. Sag. Minimal recursion semantics: An introduction. URL <http://www-csli.stanford.edu/aac/papers/newmrs.pdf>. Manuscript, Stanford University: CSLI, 1999.