

Flexible Semantic Composition with DUDES

Philipp Cimiano

Web Information Systems Group, TU Delft

1 DUDES

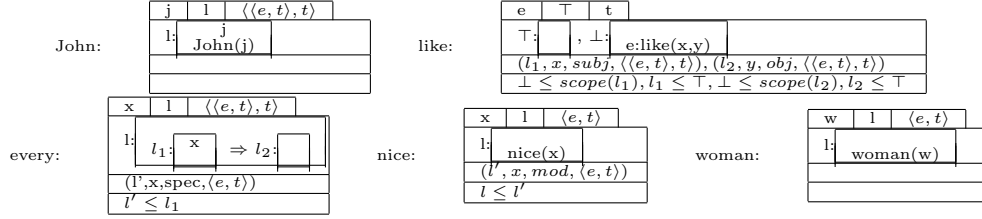
In this paper we present a novel formalism for semantic construction called DUDES (**D**ependency-based **U**nderspecified **D**iscourse **R**epresentation **S**tructures). The DUDES formalism has been designed to overcome the rigidity of semantic composition based on the lambda calculus (where the order of application is typically fixed) and provides some flexibility with respect to the direction of the dependence and with respect to the order of application of arguments. In this short paper we present the DUDES formalism and work through a simple example. DUDES bears some resemblance to the work on λ -DRT [2] and *LUDs* [1] as well as with the work of Copestake et al. [4] and represents a generalization of the formalism introduced in [3]. A detailed discussion of the relation to these formalisms is clearly out of the scope of this paper. DUDES are characterized by three main facts. First, they represent semantic dependencies and are thus inherently suitable for a dependency-based grammar formalism assuming that syntactic dependencies correspond to semantic dependencies (though the correspondence might be “inverted”). Second, they explicitly encode scope relations and are thus able to yield underspecified representations as output (in contrast to the linear logic approach for LFG [5] where different scopings correspond to different derivations). Third, there is one single operation for semantic composition which is, to some extent, order-independent (in contrast to traditional lambda-based formalisms) as well as flexible with respect to the direction of the syntactic dependency. As the name suggests, DUDES builds on DRT [6] and in particular on UDRT [7] in the sense that it relies on labeled DRSs and dominance relations between these to talk about scope. First of all, we now first formally introduce DUDES:

Definition 1 (DUDES)

A DUDES is a 7-tuple (m, l, t, U, A, S, C) consisting of

- A main discourse referent m and a distinguished label l
- The type t of the semantic structure (after inserting all arguments)
- A set U of UDRS components. These UDRS components are in essence labeled DRs [6].
- A set of argument quadruples $(l, v, rel, type)$ consisting of i) a label l (the main label of a DUDE inserted as argument, ii) the main variable v of the argument DUDE, iii) a grammatical relation and iv) a semantic type.
- A set S of scope relations between labels introduced in U .
- A set C of constraints on the arguments, e.g. including ontological constraints or selectional restrictions etc.

We now give the semantic representation of the entries for our running example: “John likes every nice woman.”:



Further, we introduce the semantic composition operation for DUDES along a dependency tree, distinguishing tree cases:

Definition 2 (Semantic Composition for DUDES)

Let (γ_1, γ_2) be an edge in some DAG (dependency tree, LTAG derivation tree or F-Structure DAG). Assume the edge is labeled with r (a grammatical function) and the semantics of the vertices γ_1 and γ_2 are given by DUDEs as follows: $\sigma_1 := (m_1, l_1, t_1, U_1, A_1, S_1, C_1)$ and $\sigma_2 := (m_2, l_2, t_2, U_2, A_2, S_2, C_2)$. Then the result of applying σ_2 to σ_1 is the DUDE $\sigma' = \sigma_1(\sigma_2) = (m', l', t', U', A', S, C')$, where we need to distinguish the following cases:

| | | |
|--|---|--|
| <p>if $(l, v, r, t_2) \in A_1$ (Complementation)</p> <p>$m' := m_1$ $t' := t_1$ $U' := U_1 \cup U_2$ $A' := A_1 \setminus \{(l, v, r, t_2)\}$ $S' := S_1 \cup S_2$ $C' := C_1 \cup C_2$ $v \equiv m_2, l \equiv l_2$</p> | <p>if $(l, v, r, t_1) \in A_2(t_1 = t_2)$ (Modification)</p> <p>$m' := m_1$ $t' := t_1$ $U' := U_1 \cup U_2$ $A' := A_1$ $S' := S_1 \cup S_2$ $C' := C_1 \cup C_2$ $v \equiv m_1(\equiv m_2), l \equiv l_1$</p> | <p>if $(l, v, r, t_1) \in A_2(t_1 \neq t_2)$ (Inversion)</p> <p>$m' := m_2$ $t' := t_2$ $U' := U_1 \cup U_2$ $A' := A_2 \setminus \{(l, v, r, t_1)\}$ $S' := S_1 \cup S_2$ $C' := C_1 \cup C_2$ $v \equiv m_1, l \equiv l_1$</p> |
|--|---|--|

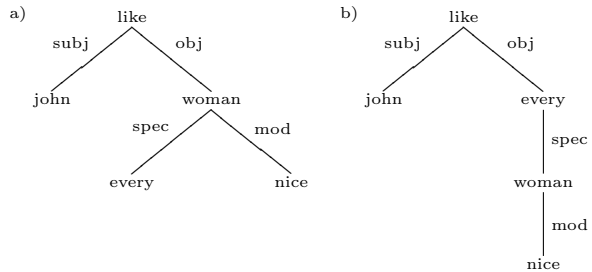
where ‘ \equiv ’ is the operation of unification between variables.

Concerning the order of application, from the definition of the semantic composition operator it follows that complements and modifiers can be applied in any order, but inversions have to be carried out at the end as

they change the mother DUDES and would thus inhibit the application of the complements and the modifiers. In the following section we show how the semantic composition operation defined above applies to a concrete example. We will also discuss that our operations still work if (some of) the dependencies are inverted.

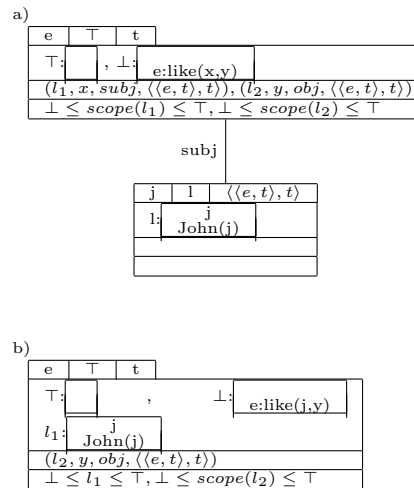
2 A Worked Example

We will consider the two following (possible) analyses for the sentence: “*John likes every nice woman.*”, corresponding to the *NP analysis* (a) and *DP analysis* (b), respectively.



2.1 Complementation

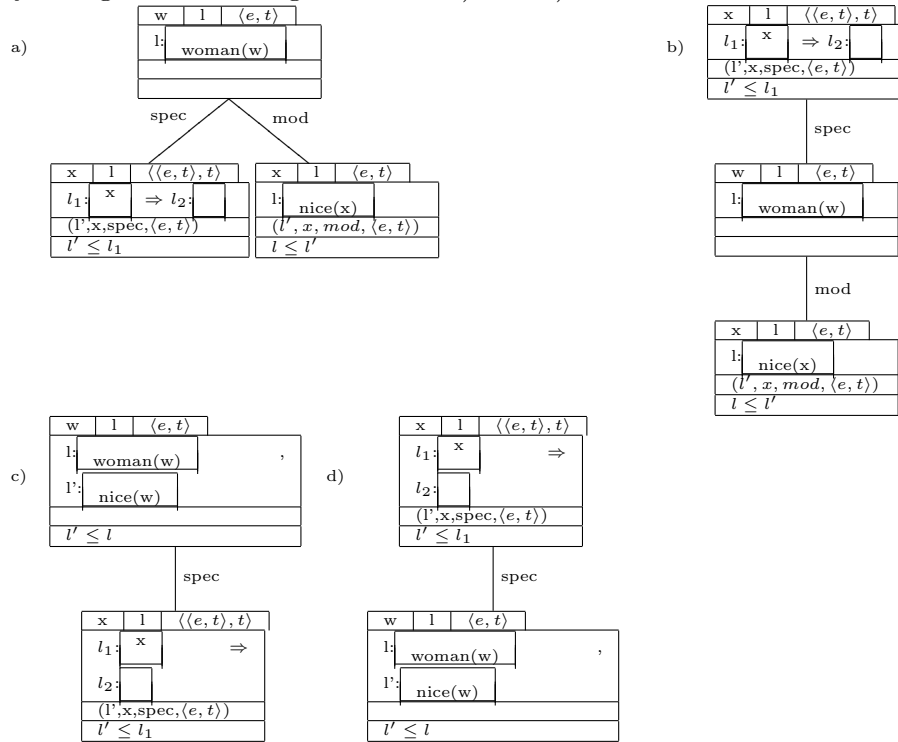
Given the dependency analysis in a) to the right, as a result of applying our semantic composition operator for the complementation case we get a DUDES where the argument has been correctly inserted, the DRS components and the scope conditions have been merged and one argument has been removed. Note that this was possible because i) the edge was labeled with the appropriate grammatical relation ‘*subj*’ and ii) the types of σ_2 and of the argument match (both are of type $\langle\langle e, t \rangle, t\rangle$). The resulting DUDES for $\llbracket \text{John likes} \rrbracket$ is shown in b) to the right. (In case of DRS conditions which are not complex, we assume that the functions ‘*res*’ and ‘*scope*’ are resolved to the identity function.)



2.2 Specification and Modification

The two possible dependency analyses for determiner/noun constructions give rise to two configurations, corresponding to a) and b) below, for the semantic composition operator. In both cases, independent of the fact whether

the determiner is the dependent or the head, we get that first the semantic representation of the adjective is applied to the one of the noun (as the modifier has to be applied before the inversion is carried out in configuration a), thus yielding the two configurations in c) and d).



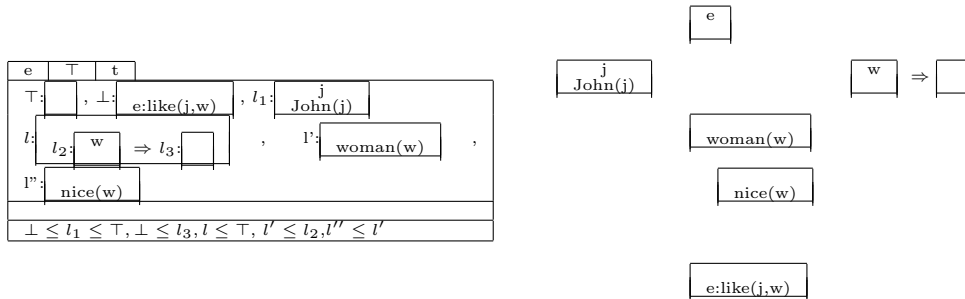
In case c) we have a case of inversion, while in case d) we have a case of complementation. Overall, in both cases we yield the following DUDES:

$$\llbracket \text{every nice woman} \rrbracket =$$

| | | |
|----------------------------|---|---|
| w | l | $\langle\langle(e, t), t\rangle\rangle$ |
| l ₁ : | w | \Rightarrow l ₂ : |
| , l: | | |
| woman(w) , l': | | |
| nice(w) | | |
| l' ≤ l, l ≤ l ₁ | | |

2.3 Result

After a further semantic composition step (case complementation), applying $\llbracket \text{every nice woman} \rrbracket$ (from Sec. 2.2) to $\llbracket \text{John likes} \rrbracket$ (from Sec. 2.1), we yield as resulting UDRS:



References

- [1] J. Bos, B. Gambäck, C. Lieske, Y. Mori, M. Pinkal, and K. Worm. Compositional semantics in verbmobil. In *Proceedings of COLING'96*, 1996.
- [2] Johan Bos, Elsbeth Mastenbroek, Scott Mcglashan, Sebastian Millies, and Manfred Pinkal. A compositional drs-based formalism for nlp applications. In *Proceedings of the International Workshop on Computational Semantics*, 1994.
- [3] P. Cimiano, A. Frank, and U. Reyle. UDRT-based semantics construction for LTAG – and what it tells us about the role of adjunction in LTAG –. In *Proceedings of the 7th International Workshop on Computational Semantics*, pages 41–52, 2007.
- [4] Ann Copestake, Alex Lascarides, and Dan Flickinger. An algebra for semantic construction in constraint-based grammars. In *Proceedings of ACL'01*, 2001.
- [5] Mary Dalrymple, John Lamping, Fernando C. N. Pereira, and Vijay Saraswat. Linear logic for meaning assembly. revised version of the (overview) paper in Proc. of the Workshop on Computational Logic for Natural Language Processing, Edinburgh, UK. 1995., 2002.
- [6] H. Kamp and U. Reyle. *From Discourse to Logic*. Kluwer, 1993.
- [7] Uwe Reyle. Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics*, 10(2):123–179, 1993.