Anders Søgaard, Center for Computational Modelling of Language:

K-structure - (a prerequisite for) an Interlingua

Abstract

This paper demonstrates how the coverage and adequacy of deep grammar-based machine translation systems can be enhanced by positing a level of *k-structure*, i.e. a level of qualia or dependent types, in the lexical semantics. This structural level provides a vocabulary for stating fine-grained semantic differences in logical form, which then can be used as an intermediate representation in translation. K-structure is a first-order language and thus also enables inferential disambiguation.

1 Introduction

There are many approaches to automatic translation of natural language. Some systems are based on probabilistic inference. Others are transfer-based, and again others are interlingual. One problem that all these systems share, and which each of them tries to solve in its own manner, is *disambiguation*, incl. syntactic ambiguities, scopal ambiguities and lexical ambiguities. Disambiguation is relevant to translation, a prerequisite for it, whenever an ambigous element α translates into two elements β and δ in the target language. Syntactic and scopal ambiguities are perhaps the more well-studied, but lexical disambiguation has far too vast a literature to present in this short a paper. Rather, the focus of this paper is on developing a simple semantic representation language that combines lexical and formal semantics in such a way that translation puzzles, which stem from lexical ambiguity or ambiguity in lexical composition, are solved. This representation language was reformulated in the Minimal Recursion Semantics architecture (henceforth, MRS). This reformulation was convenient, since it allowed an implementation in the LKB system. Case studies considered in this paper are: the semantics of spatial prepositions and prepositional linking elements in Romance compounds.

2 Representation Languages for Machine Translation

Let's start off with some truisms of transfer-based and interlingual machine translation theory:

(i) if we translate source strings into full-fletched scoped logical forms, the equivalence problem is undecidable (Shieber, 1993)

(ii) by implicit conjunction and scope elimination, a computationally simpler semantic representation is obtained (e.g. Copestake, 1995); on the other hand, we want to be able to potentially reconstruct scoped forms (for disambiguation) and not to overgenerate such forms, i.e. produce unlikely scopings

(iii) there are then established ways of restricting the translation equivalence space: reintroducing scope by handles or event branching, and specifying the event structure

In our representational language, restrictions on the equivalence space are given as command principles in event trees and as event restrictions. In the MRS architecture, the handle constraint approach is adopted. The two approaches seem more or less equivalent.

3 Some Informal Notes on Our Language

The most basic properties of our language are:

(i) it's Davidsonian (event-based)

(ii) it's anti-Anscombian (decompositional)

(iii) it's non-resource-sensitive

(iv) it's very simple

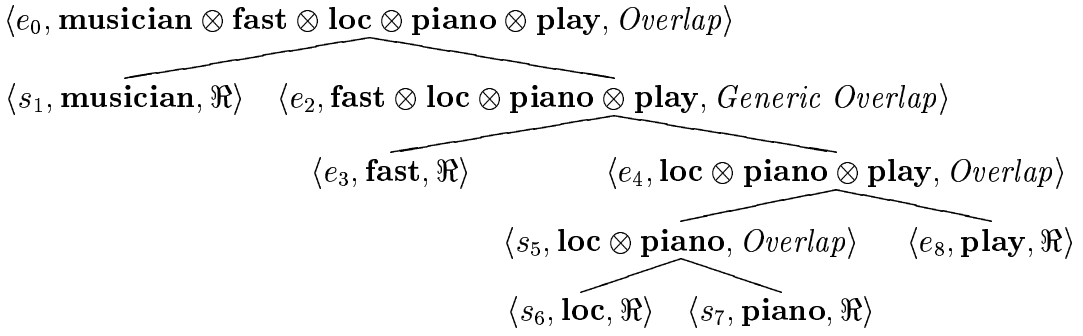
None of these properties are accidental, and we have already argued for the necessity of (iv). To give a hint at the advantages of event-based, decompositional representations and non-resource-sensitive inference, consider:

(1) (the) musician is fast at (the) piano[1]

The argument is as follows, all too brief and informal: Since *fast* modifies an event (for some arguments, see Pustejovsky, 1993), and since that event is the telic aspect of *piano* (the act of playing), an event of playing must be represented in the logical form. Our analysis is consequently anti-Anscombian, since the locative relationsship must also be represented. The preposition *by* does not allow modification of the telic aspects of the prepositional object. It only has a locative reading in this context. Not to loose the locative reading, when obtaining the telic one, our logic is to be non-resource-sensitive.

Any expression in our language is derived from elementary predications, a predicate P and some arguments e (i.e. the event variable), x and (optionally) y. The (external) arity of an elementary predication ranges from one to two. Event variables come in different flavors, represented in a type hierarchy. Conjunction is implied in logical form.

Event structure can be represented by event trees. The trees branch of as tuples $\langle \alpha, \beta, \Re \rangle$ where α and β are nodes, and $\Re$ is an event restriction. Since tuples are also nodes, the definition is recursive. Restrictions include overlap, scope, generic overlap and precedence. While $\forall \langle \alpha, \beta \rangle . Overlap(\alpha, \beta) \Leftrightarrow Overlap(\beta, \alpha)$, $\forall \langle \alpha, \beta \rangle . \neg Scope(\alpha, \beta) \Leftarrow Scope(\beta, \alpha)$. The preliminary event tree representing the partial semantics of (1) is:



[1]The determiners are ignored for clarity of exposition. Their translation is given in Table 1. In reality, the **musician** node in the event trees below comprises a binary tree with intersective daughters, **the** and **musician**. Their internal relation is one of *Overlap*, i.e. a non-scopal relation.

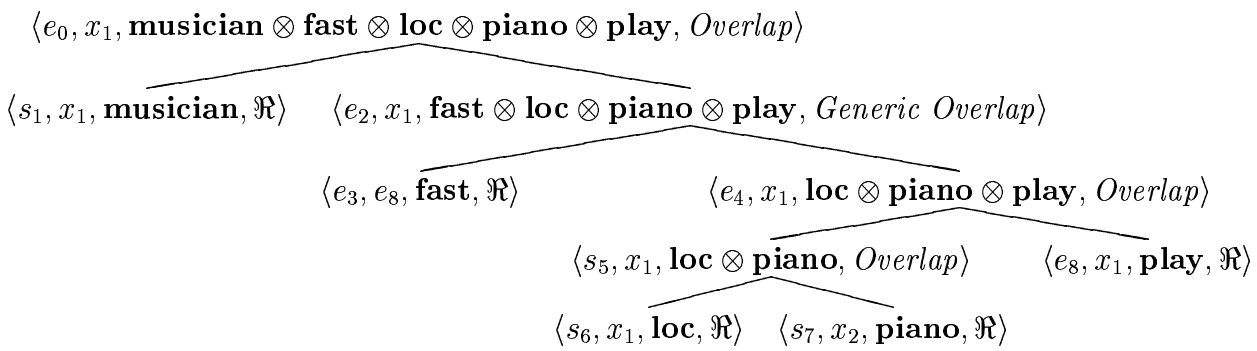
The mother node label of a tree $\langle \alpha, \beta, \Re \rangle$ is named by another 3-tuple $\langle \upsilon, \tau, \Re \rangle$ where υ is the indexed event variable, which also contain information about the general event type (at this point we have only introduced the two basic eventualities: events and states) and τ is the concatenation of the semantic types of the daughters. The well-formedness of trees is constrained by the following additional principles:

THE ENDOCENTRICITY PRINCIPLE
All event trees are headed, i.e. the general event type of the mother node is inherited from one of the daughter nodes.

THE HEAD-FINAL PRINCIPLE
All event trees are headed by their right daughters.

For composition, a projection architecture can be defined from the configurational structure into this representation language. This projection is defined in parallel with the logic that allows us to derive the subatomic dependent types, e.g. **play** from **piano**. One additional problem we have to address is the proper linking of arguments. In the preliminary event tree above, no linking is specified.

In addition to the top event variables, we thus introduce (bound) top referential indeces. An endocentricity principle constrains the distribution of top referential indeces, but notice that pairs of events and referential indeces are never constrained to co-occur up the event tree:



Only the indexed arguments are given in the trees. This information together with the arities of the predicates suffices to reconstruct traditional logical forms, e.g.:

(2) $\exists x_1.\mathbf{musician}(x_1) \wedge \mathbf{uniq\_id}(x_1) \rightarrow \Gamma e_2.[\exists \langle x_3, s_4 \rangle [\mathbf{piano}(x_3) \wedge \mathbf{uniq\_id}(x_3) \wedge \mathbf{loc}(s_4, x_1, x_3) \rightarrow \mathbf{play}(e_2, x_1, x_3)] \rightarrow \mathbf{fast}(e_2)]$

Finally, if we modify the even tree syntax a little to allow for sets of restrictions on each node, we can introduce *Scope*. Just like *Precedence*, *Scope* comes in two flavors, *Scope*$^{\rightarrow}$ and *Scope*$^{\leftarrow}$, i.e. depending on the order of restriction and nuclear scope:

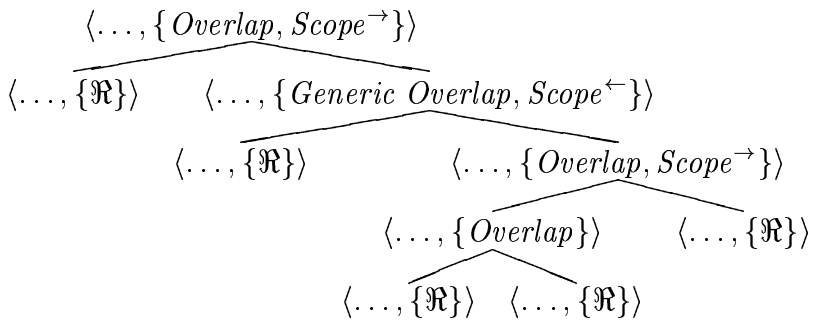


Table 1: A small fragment of English.

Constituents	**Translations**
the	$\langle s_0, x_1, \mathbf{uniq\_id}, \{\uparrow \mathit{Overlap}\}\rangle$
musician	$\langle s_0, x_1, \mathbf{musician}, \{\Re\}\rangle : \langle e_2, x_1, \mathbf{play}, \{\Re\}\rangle_t$
is	$\langle \ , \ , \ , \{\downarrow \mathit{Scope}^{\rightarrow}\}\rangle$
fast	$\langle e_0, e_1, \mathbf{fast}, \{\uparrow \mathit{Generic\ Overlap}, \uparrow \mathit{Scope}^{\leftarrow}\}$
at	$\langle s_0, x_1, \mathbf{loc}, \{\uparrow \mathit{Overlap}\}\rangle$
piano	$\langle s_0, x_1, \mathbf{piano}, \{\Re\}\rangle : \langle e_2, x_1, \mathbf{play}, \{\Re\}\rangle_t$

The advantage of this representational language may be hard to see, but these are highlighted by our inferential inventories, i.e. the composition logic. Consider the translations of the constituents of our small fragment of English in Table 1.

The structure $\upsilon_i : \upsilon_j$ indicates a possible inference from υ_i to $\upsilon_i \otimes \upsilon_j$. The subscript t is meant to indicate that the secondary tuples correspond to the telic qualia in the constituents' lexical semantics. The $\mathit{Scope}^{\rightarrow}$ in the subtree of **fast** is introduced by the inference itself. In principle, there are two ways of deriving the proper semantics, i.e. two possible inferences. For simplicity, consider only the inference on the semantic type of the prepositional object. The rest of the composition is fairly standard.

Consider the derivation of the **musician** subtree:

$$\langle s_0, x_1, \mathbf{uniq\_id} \otimes \mathbf{musician}, \{\mathit{Overlap}\}\rangle$$
$$\langle s_1, x_1, \mathbf{uniq\_id}, \{\uparrow \mathit{Overlap}\}\rangle \quad \langle s_2, x_1, \mathbf{musician}, \{\Re\}\rangle$$

Consider also the representation of the inferential structure: (The "*" indicates that this event tree is ill-formed, i.e. the e_i variable is not bound.)

$$*\langle e_0, x_1, \mathbf{fast} \otimes \mathbf{loc} \otimes \mathbf{piano}, \{\mathit{Generic\ Overlap}, \mathit{Scope}^{\leftarrow}\}\rangle \quad \rightarrow$$
$$\langle e_1, e_i, \mathbf{fast}, \{\uparrow \mathit{Generic\ Overlap}, \uparrow \mathit{Scope}^{\leftarrow}\}\rangle \quad \langle s_2, x_1, \mathbf{loc} \otimes \mathbf{piano}, \{\mathit{Overlap}\}\rangle$$
$$\langle e_0, x_1, \mathbf{fast} \otimes \mathbf{loc} \otimes \mathbf{piano} \otimes \mathbf{play}, \{\mathit{Generic\ Overlap}, \mathit{Scope}^{\leftarrow}\}\rangle$$
$$\langle e_1, e_4, \mathbf{fast}, \{\uparrow \mathit{Generic\ Overlap}, \uparrow \mathit{Scope}^{\leftarrow}\}\rangle \quad \langle e_2, x_1, \mathbf{loc} \otimes \mathbf{piano} \otimes \mathbf{play}, \{\mathit{Overlap}, \mathit{Scope}^{\rightarrow}\}\rangle$$
$$\langle s_3, x_1, \mathbf{loc} \otimes \mathbf{piano}, \{\mathit{Overlap}\}\rangle \quad \langle e_4, x_1, \mathbf{play}, \{\Re\}\rangle$$

While this presentation of the language was all too brief and informal, it is meant to indicate the basics of a language useful for chrystalizing the fine-grained meaning differences relevant to translation.

4 The Empirical Data and Their Representation

In this section, our representational and inferential language is applied to some empirical phenomena. Consider first the translation pairs below. For each prepositional construction in English, the Danish translation equivalent is given. The translation (mis-)matches highlight the need for the inferential structures discussed above:

(3) (a) the woman at the blackboard →
kvinden ved tavlen

(b) the chair by the blackboard →
stolen ved tavlen

(c) the woman in the hospital →
kvinden på hospitalet

(d) the chair at the hospital →
stolen på hospitalet

(e) the woman at the office →
kvinden på kontoret

(f) the chair in the office →
stolen på kontoret

In the English prepositional systems, certain presuppositions go with certain prepositional constructions, e.g. *?the chair at the blackboard* is ungrammatical or at least odd, since it is presupposed that the chair is somehow using the blackboard. Some of these distinctions are neutralized in Danish. Of course translation from English to Danish is then easy, but how do we make sure that the Danish prepositional constructions are lined up with their equivalents, when their presuppositional structure is not reflected at the surface level?

The simple answer in this architecture is that sortal restrictions on the dependent types of the prepositional objects will rule the odd combinations, e.g. *?the chair at the blackboard* is odd, because the greatest lower bound of **chair** and the first argument of the telic quale of **blackboard** is $\perp$. Alternatively, the odd reading can be ruled out by resolution of the output (being first-order) and relevant meaning postulates.

This is very similar to translation of polysemic compound nouns into Romance:

(4) (a) ice knife →
coltello da ghiaccio

(b) ice knife →
coltello di ghiaccio

In this case, disambiguation is again a matter of selectional restrictions on dependent types. Consider the implementation of this translational pattern below.

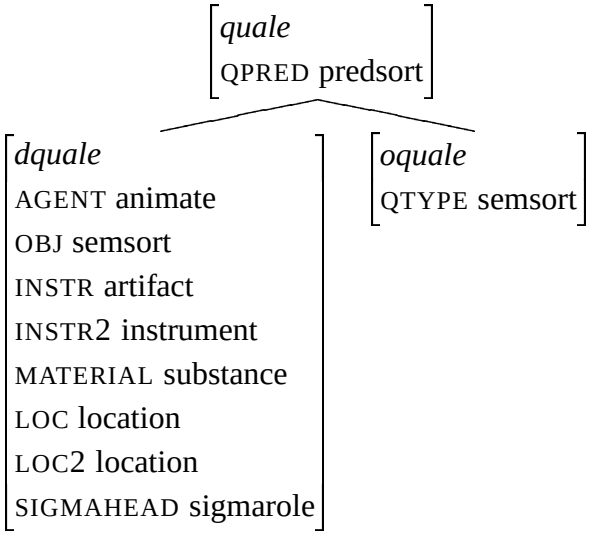
5 On an Implementation in LKB

This section demonstrates a principled implementation of the translational pattern in (4). The backbone of the analysis is the implementation of (extended) qualia structure in the LKB system.

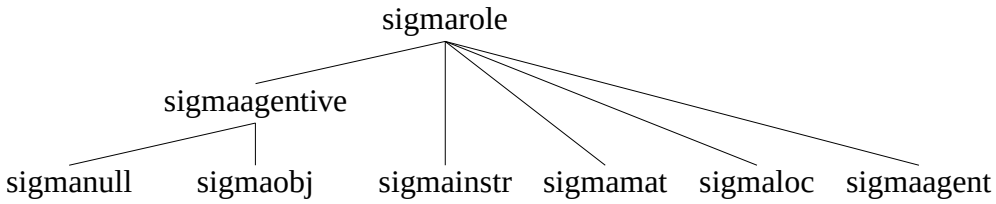
The translation of aspects of our representational language into MRS is not all one-to-one, but it allows almost similar inferential structures in a unification-based formalism. Qualia structure has a basic feature geometry:

$$\begin{bmatrix} \textit{individual} \\ \text{FORMAL } \textit{oquale} \\ \text{TELIC } \textit{dquale} \\ \text{AGENTIVE } \textit{dquale}\begin{bmatrix}\text{SIGMAHEAD sigmaagentive}\end{bmatrix} \\ \text{CONST } \textit{oquale} \end{bmatrix}$$

The values of the different quales are defined in the following type hierarchy:



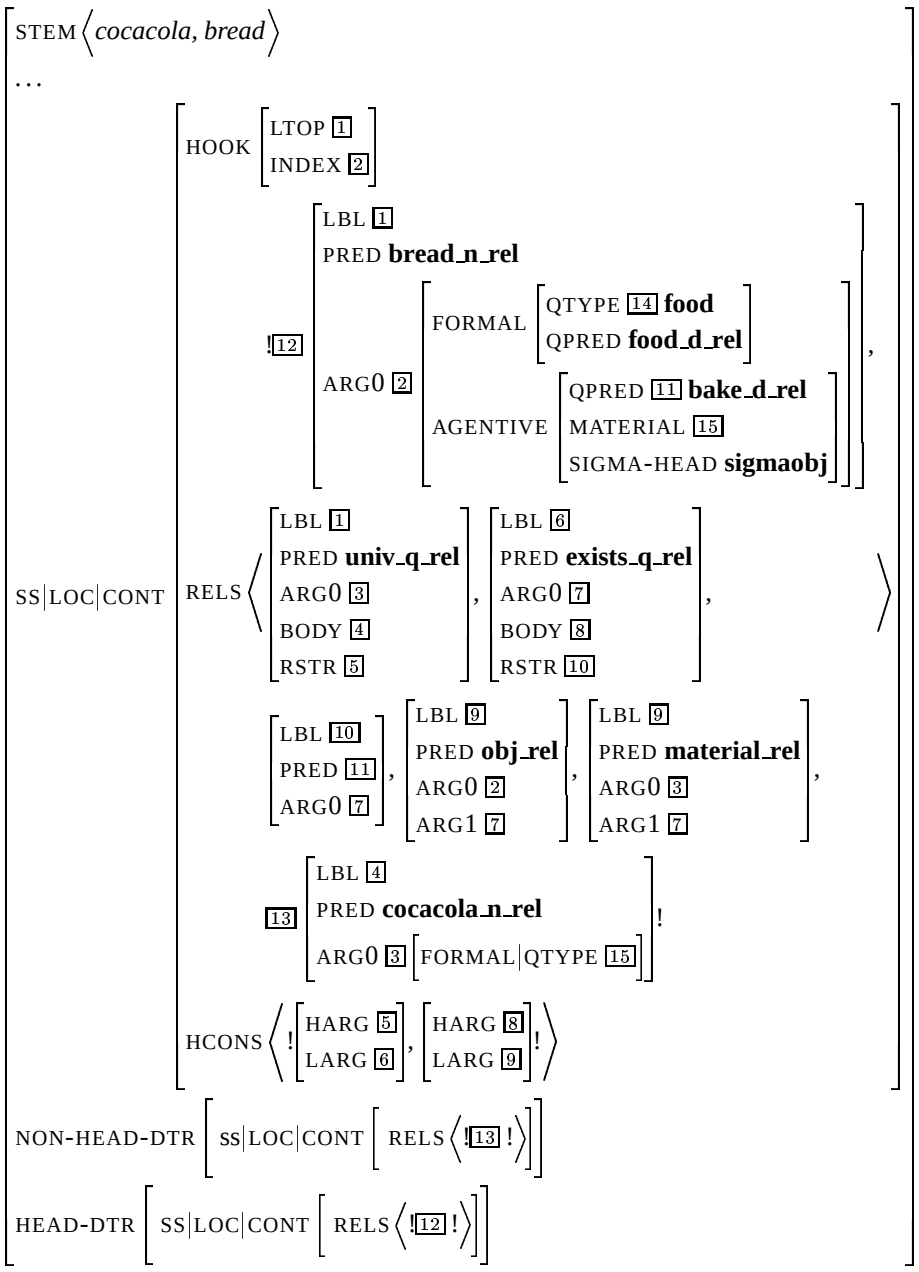
SIGMAHEAD indicates the Σ-role of a certain type in its own qualia. The possible values for this attribute are defined in this hierarchy:



With this set-up, the qualia structure is simply entered at the lexical level:

```
bread1 := nom-lxm &
[STEM <"bread">,
LANGUAGE english,
SYNSEM.LOCAL.CONT [HOOK.INDEX [FORMAL [QTYPE food,
QPRED food_d_rel],
AGENTIVE [QPRED bake_d_rel],
CONTOUR [QTYPE ellipse,
QPRED ellipse_d_rel],
TELIC [QPRED eat_d_rel,
MATERIAL nullmaterial,
SIGMAHEAD sigmaobj]],
RELS <![PRED _bread_n_rel]!>]].
```

At the constructional level, composition refers to the qualia structures. Consider the full representation of *cocacola bread*:[2]



LKB extracts from this a scoped logical form (the **intr**-predicate is contributed by the intransitive verb):

proposition_m_rel(indef_q_rel(x6, _intr_v_rel(e2, x6),
_bread_n_rel(x6) /\ univ_q_rel(x8, _coca-cola_n_rel(x8),
indef_q_rel(e14, obj_rel(x6, e14) /\ material_rel(x8, e14),
bake_d_rel(e14)))))

[2]In the current version of the MRS formalism, event variables are unbound, i.e. constants; we adopt this design for clarity of exposition; for some discussion, see Partee (2000).

Can we arrive at the same logical form for the Italian equivalent in a principled manner? Our easy solution to this problem is to assign a null semantics to the linking element. Its contribution is simply to constrain the possible compound constructions that can apply. It constrains the relevant qualia to be agentive, and the value of SIGMAHEAD to be *sigmaobj*, while the Σ-relation is constrained to be $\lambda x_1.\mathbf{material\_d\_rel}(x_1)$. Consequently, the LKB output for *pane di cocacola* is equivalent to the output presented above:

proposition_m_rel(indef_q_rel(x6, _intr_v_rel(e2, x6),
_bread_n_rel(x6) /\ univ_q_rel(x9, _coca-cola_n_rel(x9),
indef_q_rel(e14, obj_rel(x6, e14) /\ material_rel(x9, e14),
bake_d_rel(e14)))))

The compound *ice knife* has two interpretations. One of these is provided by this here construction, while the other relies on telic qualia. Since the prepositional linking elements *di* and *da* further specify the parse input, it is only natural that ambiguity is reduced. Only the first construction applies to strings of the form: α *di* β.

What's lost in conversion into MRS The inferential structures stated in the first section do not have an exact correspondence in the MRS language. While our language admits abstractly defined inferences, e.g. inference of any eventive dependent type, the MRS formalism and its implementation in typed feature structure logic force us to explicitly state each such kind of inference. In other words, the specific dependent types must be stated in the latter language. In addition, the temporal structure is not expressed in MRS.

6 Conclusion

The present paper documents the basics of a formal language, which we tentatively have called *k-structure* (Søgaard, 2004), and which was designed to support deep grammar-based machine translation. More specifically, it was designed to reason about dependent types in a simple fashion. Instead of a direct implementation of this language, the paper also presents a translation of k-structure into MRS, a language partially developed on the LKB platform. The implementation was succesful and exemplifies automatic translation of polysemic compound nouns in Germanic into disambiguated prepositional constructions in Romance. Since the output of both systems can be interpreted as first-order logical forms, knowledge-based disambiguation can easily be performed by theorem provers and model builders.

References

Copestake, Ann. 1995. Semantic transfer for Verbmobil. Verbmobil report 93.

Partee, Barbara. 2000. Some remarks on linguistic uses of the notion of "event". In Carol Tenny and James Pustejovsky (eds.), Events as grammatical objects. Stanford: CSLI.

Pustejovsky, James. 1993. Type coercion and lexical selection. In James Pustejovsky (ed.), Semantics and the lexicon. Dordrecht: Kluwer.

Shieber, Stuart. 1993. The problem of logical-form equivalence. Computational Linguistics 19: 179-190.

Søgaard, Anders. 2004. Compound theories and linguistic diversity. In Z. Frajzyngier et al. (eds.), Language theories and linguistic diversity. Amsterdam: John Benjamins.