Large Scale Deep Grammars – a success, and a long way to go

Lars Hellan

Research Group in Digital Linguistics, Trondheim

LACompLing2017, Stockholm, August 16, 2017

Large Scale Deep Grammar (LSDG)

By a *'large scale deep grammar'* we may understand a computational grammar which performs formal morpho-syntactic and semantic analysis at a level of specification of linguistic frameworks such as HPSG, CCG, LFG, …

LSDGs are expected to sustain applications processing free text input with expert execution, such as grammar correction and high accuracy text interpretation. The advantages of LSDGs thus lie in execution of tasks related to freely chosen, ever 'new', text. The perspective taken in this talk is not this aspect of the grammars, however, but that of the intrinsic *linguistic* interest of LSDGs.

The perhaps most interesting linguistic aspect of LSDGs is their capacity as 'complete functions' from data to analysis, and thus as symbolic models of 'total grammars' as such. Through them, one should be able to study on the one hand patterns of 'large' structures of the grammar, and on the other hand the integration of more fine-grained patterns in the large structures. The circumstance that they interact in one overall system is a higher-order fact by itself, and an LSDG *could* be a tool to investigate it.

Large Scale Deep Grammar (LSDG)

A lexical or grammatical pattern can of course be studied separately, 'out of the grammar', but that loses the perspective that for each pattern to have its properties, it is by virtue of being *part of* the grammar, thus *'in the grammar*' – it could be a bit like organs of a deep water fish exhibiting effects of the constant pressure under which it lives.

In this we are not invoking possible physical limitations carried by the species using language as conditioning factors – we are just talking about the 'rule' system that can be abstracted from the collective behaviour constituting a language.

It is clear that computational small grammars are instructive for learning and mastering a formal framework; and we know that grammars are inherently very large, and that if we want to give a comprehensive, concise formulation of it, encoding this in a computational shape is excellent for *consistency* ensurance, and thereby an excellent medium for cumulative *description*.

Still our perspective here is more whether a grammar so created can inform us about particular linguistic research questions, and whether linguistics on the whole is better conducted when making use of LSDGs. This will be our Leitmotiv.

Large patterns

We may note that our grammar has 90,000 lexical entries, of which 12,000 are verbs, 12,000 are adjectives, and 65,000 are nouns;

a certain percentage of words of each class have irregular inflections; the features for which the words inflect constitute certain (partly overlapping) sets;

some of the features are also expressed among the ’constant’ words (the remaining 1000 – the ’grammatical words’).

Aspects of semantics pervade all words, but perhaps differently according to word class – does a ’meaning’ predict what kind of word it comes out as?

For instance, what we call ’argument structure’ is a feature mainly of situation-expressing words, thus mainly but not exclusively verbs. How many grammaticalized argument structures are there, and what types of structures can one establish for them?

Parameters of argument structure

- syntactic argument structure, i.e., whether there is a subject, an object, a second/indirect object, etc., referred to as grammatical functions, and the formal categories carrying them;
- semantic argument structure, that is, how many participants are present in the situation depicted, and which roles they play (such as ‘agent’, ‘patient’, etc.);
- linkage between syntactic and semantic argument structure; here also belong identity relations, part-whole relations, etc., between arguments;
- aspect and Aktionsart;
- within certain domains, type of the situation expressed, in terms of some classificatory system.

Valence questions: Frame pods

A verb can often be seen as taking more than one valence frame, and we will refer to a set of frames taken by a given verb as a *frame pod*. From existing estimates, we may say that about half of the verbs in a language take only one frame, the others mostly two or three, but the number can be up to 20 for certain types of verbs, given fairly conservative criteria for what can be recognized as parts of a frame.

In a given pod, we thus have one and the same verb V occurring with more than one frame, so that the pod abstractly speaking is a set of pairs <V, Frame1>, <V, Frame2>, <V, Frame3>, etc.; we may refer to each such pair as a *val*(ency)-*instance* of the verb, and the verb V by itself as the *host* relative to that pod. By rough estimate, from a totality of verbs of a language, if one assigns the possible pods to all of them, the total number of val-instances will be about 30-40% higher than the number of verbs-per-se.

Valence profiles

One can also speak of the possible *verb frames* of a language by themselves, then using notions like ‘intransitive’, ‘transitive’, ‘ditransitive’, etc.; these notions constitute a familiar dimension of classification, although the number of such frames, given the same criteria as alluded to above for what can be recognized as part of a frame, may easily amount to between 250 and 300 for well-investigated languages, with up to 10,000 val-instances recoded (and thus around 7000 verbs-per-se, or ‘pod hosts’). These frames are not the same from language to language (although there is of course overlap, and more so the closer the languages), and so we may refer to the set of verb frames obtaining for a language as the *valence profile* of that language. Out of such a profile and a verb inventory, it is then straightforward to make overviews of which frames take which verbs and how many verbs, which frames co-occur within the same frame pods, and similar figures.

We show a system suited for such overviews, called *Construction Labeling* (Hellan and Dakubu 2010, Dakubu and Hellan 2017); it at the same time has an interface to the grammar formalism itself.

'Construction Labeling' as system for encoding valence profiles

Below are small excerpts from about 300 frame types in a valence profile using Construction Labeling code(https://typecraft.org/tc2wiki/Valence_Profile_Norwegian) :

v-tr-suAg_obAff-AFFECTING

v-tr-suAg_obAffincrem-COMPLETED_MONODEVMNT

v-tr-suAg_obTh-REPRESENTING v-tr-suAg_obThincrem-COMPLETED_MONODEVMNT

v-tr-suSens_obTh-SENSING

v-tr-suAg_obExp-CAUSED_EMOTION

v-tr-suTh_obPoss-POSSESSION

v-tr-suTh_obMal-MALEFACTION

v-tr-suTh_obExp-CAUSED_EMOTION

v-tr-suTh_obWeightunit-WEIGHING

…

v-tr-obDECL-suSens_obThsit-SENSING

v-tr-obDECL-suCog_obThsit-COGNITION

v-tr-obEqInf-suAg_obThsit-ACTIVITY

…

'Minimal construction units' (*MCUs*)

The string units of CL reside in a collection of *minimal construction units* (*MCUs*), exemplified:

Unit	*Interpretation ('X' a construction, such as a sentence)*
v	*X* is headed by a verb
tr	*X* is transitive (i.e., has two arguments)
ditr	*X* is ditransitive (i.e., has three arguments)
suAg	*X* has a subject carrying the role *Agent*
obAffincrem	*X* has an object carrying the role *Incrementally Affected*
obEndpt	*X* has an object carrying the role *Endpoint*
ob2Mover	*X* has a second object carrying the role *Mover*
ACCOMPL	*X* expresses the Aktionsart *Accomplishment*
PLACEMENT	*X* expresses the situation type *Placement*

Correspondence between MCUs and AVM format

`v-tr-suAg_obAffincrem-ACCOMPL`

A sentence instantiating this type would be *The girl ate the cake*, with the properties :

v	'is headed by a verb'
tr	'is transitive (i.e., has two nominal arguments)'
suAg	'has a subject carrying the role *Agent'*
obAffincrem	'has an object carrying the role *Incrementally Affected'*
ACCOMPL	'expresses the Aktionsart *Accomplishment'*

The following is an AVM expressing this content:

$$\begin{bmatrix} \text{HEAD } \textit{verb} \\ \text{GF} \begin{bmatrix} \text{SUBJ} \left[\text{INDX } \boxed{1} \left[\text{ROLE } \textit{agent} \right] \right] \\ \text{OBJ} \left[\text{INDX } \boxed{2} \left[\text{ROLE } \textit{aff - increm} \right] \right] \end{bmatrix} \\ \text{ACTNTS} \begin{bmatrix} \text{ACT1 } \boxed{1} \\ \text{ACT2 } \boxed{2} \end{bmatrix} \\ \text{AKTRT } \textit{accomplishment} \end{bmatrix}$$

MCUs as unifiable Typed feature structures

`v-tr-suAg_obAffincrem-ACCOMPL`

v - - -

$$\left[\text{HEAD verb}\right]$$

tr - - -

$$\left[\begin{array}{l}\text{GF}\ \left[\begin{array}{l}\text{SUBJ}\ \left[\text{INDX}\ \boxed{1}\right]\\ \text{OBJ}\ \left[\text{INDX}\ \boxed{2}\right]\end{array}\right]\\ \text{ACTANTS}\ \left[\begin{array}{l}\text{ACT1}\ \boxed{1}\\ \text{ACT2}\ \boxed{2}\end{array}\right]\end{array}\right]$$

suAg - - -

$$\left[\text{GF}\ \left[\text{SUBJ}\ \left[\text{INDX}\ \left[\text{ROLE agent}\right]\right]\right]\right]$$

obAffincrem - - -

$$\left[\text{GF}\ \left[\text{OBJ}\ \left[\text{INDX}\ \left[\text{ROLE aff-increm}\right]\right]\right]\right]$$

ACCOMPL - - -

$$\left[\text{AKTR accomplishment}\right]$$

Using MCU combinations

The code allows one to convert full valence profiles into typed feature structures corresponding to grammar-internal encodings of the various construction types. The MCUs are ’universal’, applying across languages, whereby their different combinations can be searched and calculated in a suitable format. Conversely, LSDGs can be converted into the CL format and thereby compared with other grammars.

If a valence profile is connected to an Interlinear Glossed Text (IGT) corpus for a language, then Grammar Induction can be done by combining induction from the MCU strings and the IGT.

The next slide shows a snippet of an IGT (for Ga) and the pairing of snippets from an XML representing it with grammar code specifications, whereby both lexical and inflectional information is imported into the grammar supplementing the syntax and semantics specifications possibly coming from the valence string conversion.

Word	etee
Morph	e | tee
Meaning	| go
Gloss	PERF |
POS	V

```
<word id="30409" text="etee" citation="etee">
    <pos>V</pos>
    <morpheme id="46593" text="eì">
        <gloss>PERF</gloss>
    </morpheme>
    <morpheme id="46594" text="tee" meaning="go"/>
</word>
```

```
tee-v :=                    v-lxm & [ STEM <"tee">,  ACTNTS.PRED tee_rel ].

verb-Perf_irule :=          %prefix (* e) word & [ ASPECT perf, INPUT < v-lxm > ].
```

Induction of 'meta'-grammar

In such a setting, one can also induce what may be called a *meta-grammar*. Here we enter not the actual words and morphemes of the language, but the *gloss* versions of these items, as they are reflected in the IGT, so that a relevant string from an IGT including the part in the given example can be

man DEF goPFV.

Thus, what we import from the IGT is not actual morphs but their glosses:

go_v := v-lxm & [STEM <"go"> , ACTNTS.PRED go_rel].

verb-PFV_irule := %suffix (* PFV) word & [ASPECT perf, INPUT < v-lxm >].

*For an implementation, see **TypeGram** (https://typecraft.org/tc2wiki/TypeGram)*

Verb classes

We have thus indicated how a grammar's internal formalism can be converted to more generally readable formats, and – as 'grammar induction' – vice versa. That allows one to project aspects of the grammar onto formats where the aspects can be more efficiently investigated. An area of research around valence is for instance that of *Verb Classes* (also called 'Valency Classes'), conducted in language typology as in the project/database *ValPaL*, with the valence frames of 80 verbs across 30 languages, and at a large scale, in a more NLP oriented direction, the English *VerbNet*,with 6340 verbs divided into 273 verb classes, at smaller scales for many languages. The topic of these investigations is what distinct verbs in the same frame pod can have in common, both inside and across languages, guided by the assumption that shared pod membership between verbs is correlated with meaning similarity, phrased in such terms as 'verb of motion', 'verb of perception' and the like – going back (in slightly different terms) to the work of Levin 1993.

Verb classes

In the context of a large grammar, establishing a concise system for representing the possible frame-pod types, the sets of verbs belonging to each frame pod type (i.e., each verb class), a measure of distance between the verb classes (in terms of membership of each class), and candidate semantic properties distinguishing each class, with semantic types and subtypes reflecting subsets of the defining sets, will constitute an interesting task, for which a concise representation of valence frames and valence profiles is then useful.

Creating a valence corpus from an LSDG

We now consider another task of conversion related to a grammar and what it says about valence. This time, however, the conversion does not pertain to the grammar internal code, but to the code used in parse results delivered by the grammar, thus 'outside' of the grammar as such.

We show how one can generate a valence corpus of Norwegian (Bokmål) from a deep grammar using the Leipzig Corpus Collection (LCC; Goldhahn et al. 2013). The corpus is presented in the form of IGT (interlinear glossed text) augmented by valence information. As our deep parser we use the computational grammar Norsource (Hellan and Bruland 2015) while our online IGT repository is TypeCraft (Beermann and Mihaylov (2014). The purpose is twofold: (i) making the grammatical information encoded in a deep parser more readily available, and (ii) facilitating a further integration of a deep parser, an online linguistic workbench, and a large corpus of text which together stand for the linkage of linguistic analysis and existing datasets to larger corpus resources. In the larger picture, it again instantiates how grammar information can be utilized for specific purposes, and this time this is accomplished through the grammar as a processor of text, thus in its 'dynamic' capacity. A trial version is described at:

https://typecraft.org/tc2wiki/Norwegian_Valency_Corpus

Illustration of the produced Valence + IGT 'normal form'

String: Jeg vet at hun forbauset Ola

Free translation: I know that she surprised Ola (Not generated from the grammar)

Morph:	Jeg	|vet	|at	|hun	|forbause	|t	|Ola
Citation:	|	|vite	|	|	|forbause		|
	|1.SG.NOM	|PRES	|DECL	|3.SG.FEM	|	|PRET	|
	|PN	|V	|COMP	|PN	|V		|Np

vet: SAS: NP+Sdecl
FCT: transWithSentCompl
ConstructionLabel: v-tr-obDECL

forbauset: SAS: NP+NP
FCT: transitive
ConstructionLabel: v-tr

NorSource ('Norwegian HPSG Resource Grammar')

As a so-called *Deep Computational Grammar*, NorSource sustains a *generic* parser (not restricted with regard to style of text or domain of use) representing wide lexical coverage, encoding linguistically well motivated morpho-syntactic and semantic analyses of nearly all aspects of the grammar, and applying this knowledge in the parsing process such that every parse reflects this knowledge. NorSource has as its formal and theoretical framework *Head-Driven Phrase Structure Grammar* (*HPSG*) (Pollard and Sag 1994, Sag et al. 2003), using the *LKB platform* (Copestake 2002), which is a general platform with the format of typed feature-structures (TFS), and has integrated in it a format of semantic representation called *Minimal Recursion Semantics* ('MRS'; cf. Copestake et al. 2005). NorSource was started in 2001 and is still being maintained and developed.

Online access, for description: http://typecraft.org/tc2wiki/Norwegian_HPSG_grammar_NorSource .

Webdemo: http://regdili.hf.ntnu.no:8081/linguisticAce/parse

The NorSource code files are downloadable GitHub, and from: http://www.nb.no/sprakbanken/show?serial=sbr-32&lang=en

The system LKB as such can be downloaded from http://moin.delph-in.net/LkbTop.

From parser to valence corpus

Whenever a sentence is parsed, this entails – given the HPSG design for full sentence parsing - that each verb in the sentence has been assigned one of the valence frames defined in the grammar for that verb, which means that every pairing of a verb with a valence frame in the parse result, and in the import to TC, is a pairing declared as correct by the grammar. In that sense, the valence information supplied for each successful parse is accurate.

Valence corpora are often built manually, or by statistical methods where hand annotation plays a crucial role (E-Valbu, Walenty, Vallex, …). In some cases valence corpora, possibly in conjunction with tree-banks, are used in the construction of computational grammars (cf., Osenova, (2011), Patujek and Przepiórkowski (2016)). Here we go the opposite way, exporting information from a deep grammar to an IGT corpus. Some steps in the procedure will now be indicated.

Using the grammar's parse tree as basis for constructing XML

Jeg vet at hun forbauset ordføreren 'I know that she surprised the mayor'

```
head-subject-rule
 jeg_perspron
  jeg
 head-verb-inf-or-s-comp-rule
  pres-infl_rule
   vite_subord_vlxm
    vet
   head-complementizer-comp-fin-rule
    at_subord
     at
    head-subject-rule
     hun_perspron
      hun
     head-verb-comp-rule
      pret-nonfstr-et_infl_rule
       forbause_tv_vlxm
        forbauset
      sg_def_m_final-full_irule
       sg-masc-def-noun-lxm-lrule
        ordfører_n_masc_nlxm
         ordføreren
```

Information from the parse tree for valence extraction

We assign a valence value to every verb occurrence in a sentence. For *vet* a look-up in the verb lexemes file establishes that the identifier in question carries the type *v-tr-obDECL* (cf. the simplified view of a verb entry in (a)), and look-up in a file establishing correspondences between the CL code and the SAS and FCT codes yields (b).

a. vite_subord_vlxm := v-tr-obDECL

b. v-tr-obDECL =>

SAS: "NP+Sdecl";

FCT: transWithSentCompl

From these correspondences the following part of the Figure is established:

vet: SAS: NP+Sdecl

FCT: transWithSentCompl

ConstructionLabel: v-tr-obDECL

The files in which these look-ups are made count 12,000 entries corresponding to (a), and about 400 conversions corresponding to (b).

Information from the parse tree for POS and GLOSS extraction

NorSource has a lemma-based lexicon, which means that inflectional processing is done via ‘rules’, stated in a form exemplified below (for verbs 22 such rules, for nouns 28, and for adjectives 38).

```
pret-nonfstr-et_infl_rule :=
%suffix (e a) (e et) (es es) (es edes)
infl-pret-verb-word &
[ARGS <[ INFLECTION nonfstr-et ]>].
```

This rule is mentioned by name in the tree for *forbauset*, reflected in the lines

```
pret-nonfstr-et_infl_rule
 forbause_tv_vlxm
  forbauset
```

stating that the form *forbauset* has been derived from the lemma form *forbause* by the application of this rule.

Information from the parse tree for POS and GLOSS extraction

The appropriate GLOSS tag in TC will be PRET, and this is assigned through the mapping rule below to the GLOSS line in TC:

pret-nonfstr_infl_rule = PRET

There are altogether 75 mapping rules from Norsource inflection rules to TC GLOSS tags. Most of them apply simply to rule names, more examples for verbs are given below:

ppart-finalstr-dd_infl_rule = PRF

s-passive_s_infl_rule = PRES.PASS

pl_def_m-or-f_light-e_irule = PL.DEF

sg_def_n_light-e_irule = SG.DEF.NEUT

Information from the parse tree for POS and GLOSS extraction

GLOSS for constant words:

fordi_comp = CAUS

idet_prep-time = TEMP

mer_cmpar-reg = CMPR

seg_refl = 3P.REFL.ACC

en_indef-art = SG.MASC.INDEF

POS for words according to entry suffix, or to word as a whole (constituting most of the 472 mappings to POS tags):

nlxm = N

alxm = ADJ

vlxm = V

dirtel-end-p = PREPdir

reg-p-loc = PREPplc

mer_cmpar-mass = QUANT

Assessments

The quality of the valence information depends on the quality of the deep parser, that is, a deep parser combines syntactic and semantic parsing with the recognition of predicate-argument structure, and our valence corpus therefore will be only as good as the parser is in handling these grammatical dependencies. Moreover the quality of the corpus depends on the convervsion itself which is not without complexity, so that mistakes could arise, and, per the automatic design, 'infect' a large number of sentences.

Yet an obvious advantage of the method is that, once analyses are deemed plausible, one can in relatively little time obtain a comprehensive valence corpus.

A valence resource should also include a valence lexicon, where each verb is specified for all the frames in which it can occur, preferably with links to selected examples from the corpus, or to the corpus seach interface. Such a facility is not yet included in TC, but is, apart from corpus links, independently available in *MultiVal*.

MultiVal *– a Multilingual Valence database*

Web demo: ttp://regdili.hf.ntnu.no:8081/multilanguage_valence_demo/multivalence

Guidelines: http://typecraft.org/tc2wiki/Multilingual_Verb_Valence_Lexicon

The system hosts 4 languages, with altogether 50 000 verb entries, with valence frames classified in a uniform system. The languages hosted are:

Bulgarian *(lexicon import from BURGER, the Bulgarian Matrix grammar)*

Ga *(lexicon import from GaGram, the Ga Matrix grammar, whose lexicon is in turn imported from a ToolBox lexicon of Ga, created by M.E.Kropp Dakubu)*

Norwegian *(lexicon import from NorSource , the Norwegian Matrix grammar)*

Spanish *(lexicon import from SRC , the Spanish Matrix grammar)*

For documentation per February 2014 (before Bulgarian got added), see Hellan et al., LREC 2014.

The web system is stable, and steadily used, but not massively. What such a resource needs in addition is systematic semantic information, and linkage to valence corpora.

Tying up so far

We have described various types of interaction between an LSDG and research-oriented resource building, partly residing in ways of accessing the grammar code itself, partly in accessing the results of the grammar's parses. The latter aspect of an LSDG is also what is used in, e.g., automatic translation and grammar correction, so in both respects we are in domains that wouldn't be available unless there was a computational grammar. We now turn to the wider question whether linguistics on the whole is better conducted when making use of LSDGs.

LSDGs actually do not play much of a role relative to 'normal' linguistics – for most languages the construction of an LSDG would not be considered even as a remote goal, and for those languages where an LSDG does obtain, the grammar as such is hardly used to inform active linguistic research, nor has active linguistic research among its foci to inform the LSDG. This holds even for frameworks specifically designed to sustain computational grammars.

This could be attributed to the complexity of the infrastructure needed to produce an LSDG; it is also a fact that the linguistic content of an LSDG is the product of a chain of analytic decisions the totality of which would hardly be shared by all colleagues of the linguist(s) in the development team. Thereto comes the circumstance that the development of an LSDG is like a strategy game, where all aspects must be developed in concord, so that no aspect can be entered into too deeply compared with other aspects.

'Fine-grained' analysis

To address this concern, we should look not only at actual LSDGs, but also on the formalisms by which they work: a formalism which sustains a functioning LSDG is obviously a respectable formalism, to be given a central place in linguistics. Staying within the formalism so far referred to, we will substantiate the point by considering the integration of fine-grained aspects of analysis in the larger structures. Essential among them is the ability to assign concise analyses to 'irregular' constructions or irregular aspects of constructions such as 'selected' prepositions and adverbs, 'bare' nominalizations, light verb constructions, and much more. While so-called 'shallow' methodologies can make lists of all the constructs falling within such a category and develop parsers that regonize whether a given pattern belongs to such a list, an LSDG in addition has to calculate the exact argument structure and semantics, on top of the morpho-syntax. This is not possible unless the 'regular' part of the grammar already handles the required parameters, and offers an analytic design where the 'irregular' cases are accomodated in a purely monotonic fashion, i.e., by filling in added specifications in slots already defined.

'Fine-grained' analysis in TFS

In a grammar formalism using TFS, 'added specification' can reside either in elaboration of values of given attributes, in introduction of new attributes, or in addition of types relative to defined type hierarchies. The former two come into play in the analysis of *Light Verb Constructions (LVCs)* in Norwegian, the latter two in the analysis of *Bare Nominalizations* in Norwegian (and probably many other languages), as we illustrate next.

Light verb constructions

One characteristic of so-called *Light Verb Constructions* (*LVC*s) is that they unfold, mostly over a sequence '*Subject* **V (P) N'**, a content that could in principle be carried by some verb ***V*** alone, and where the **N** of the sequence carries the main part of the content, hence the term 'light' for the role of the verb. The **N** thus expresses a situational content, often being 'de-verbal', and a typical role of (the 'light' verb) **V** in the LVC is to connect its *subject* to this situational content as a *role* bearer, and possibly add *aspectual* and *viewpoint* content to the situational content expressed by **N**. (a, b, c) are examples from Norwegian; (d, e, f) are corresponding sentences where a related verb alone carries the relational content:

a. Han fant behag i innspillingen *He found pleasure in the recording*

b. Han gjorde bruk av grammofonen *He made use of the gramophone*

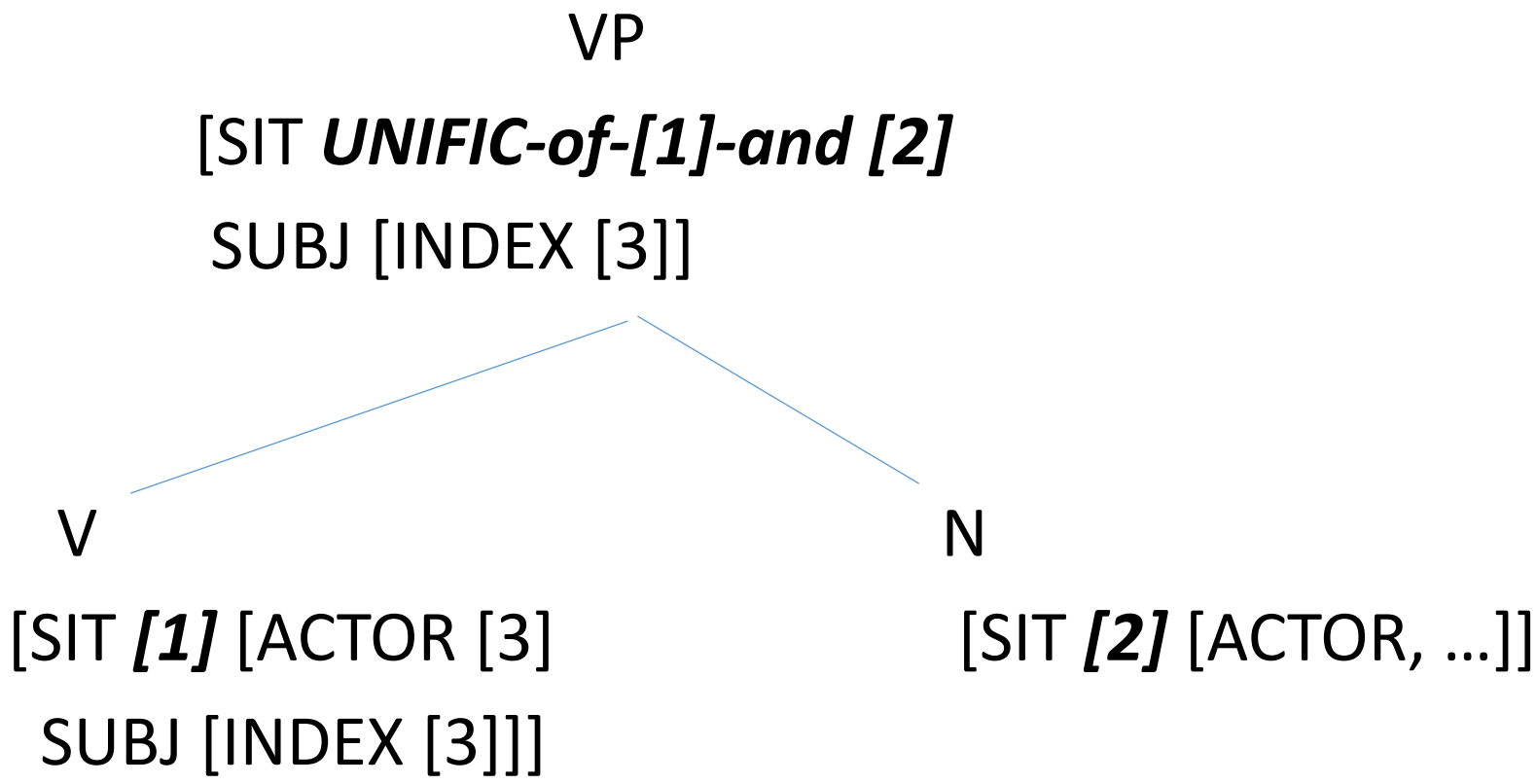
c. Hun stilte saken i bero *She suspended the case*

d. Innspillingen behaget ham *The recording pleased him*

e. Han brukte grammofonen *He used the gramophone*

f. Saken beror T*he case is suspended*

Light verb constructions – the need for merging content



Light verb constructions – situation merger, and selection

Lide ‘suffer’ is among the very few ‘light’ verbs that can combine with *nederlag* ‘defeat’, and vice versa. To build such a circumstance into the combination formalism, not only POS and SIT of the complement must be specified in the verb’s valency frame, but also a sign-specific identification of the object noun. Schematically, this will look as in (1) for the sign for *lide* (where the attribute ‘KEY’ serves for sign ‘identity tag’; note that SIT identity concerns a PATIENT role). Given that the noun equally much selects the verb, one may conceive of the selection construed the opposite way, as schematically indicated in (2), where the verb is introduced by the attribute GOVERNOR

(1)

$$\begin{bmatrix} \text{ORTH} \langle \text{"}lide\text{"} \rangle \\ \text{HEAD} \left[verb \left[\text{KEY } lide\_lvc \right] \right] \\ \text{COMPS} \left\langle \begin{bmatrix} \text{HEAD } noun \left[\text{KEY } nederlag\_lvc \right] \\ \text{SIT } \boxed{1} [\text{PATIENT}] \end{bmatrix} \right\rangle \\ \text{SIT } \boxed{1} \end{bmatrix}$$

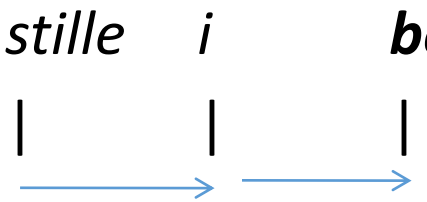
(2)

$$\begin{bmatrix} \text{ORTH} \langle \text{"}nederlag\text{"} \rangle \\ \text{HEAD} \left[noun \left[\text{KEY } nederlag\_lvc \right] \right] \\ \text{GOVERNOR} \left\langle \begin{bmatrix} \text{HEAD } verb \left[\text{KEY } lide\_lvc \right] \\ \text{SIT } \boxed{1} [\text{PATIENT}] \end{bmatrix} \right\rangle \\ \text{SIT } \boxed{1} \end{bmatrix}$$

Light verb constructions - selection

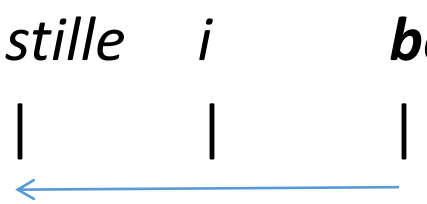
While the previous case opens for either direction of selection, the noun bero as LVC object can only occur with *stille i*, therefore the construal (b) here will seem the only option:

a.



Government of preposition and noun *by the verb*

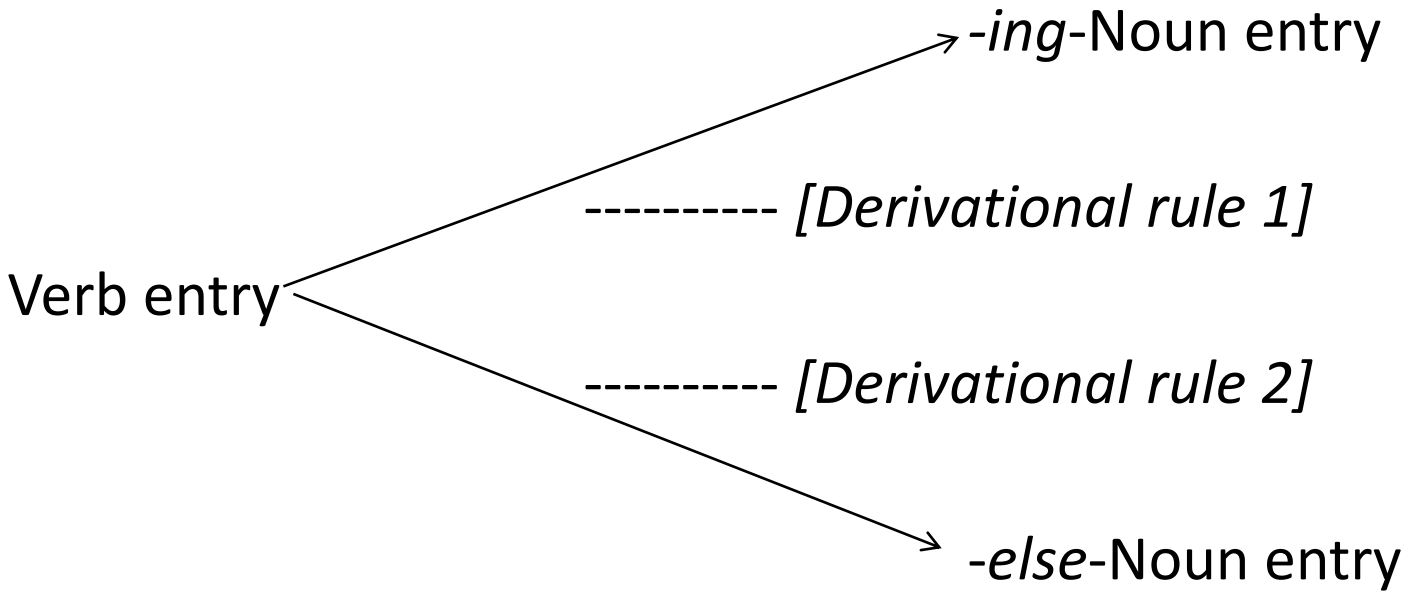
b.



'Selection' of preposition and verb *by the noun*

Bare Nominalizations

We next consider Bare Nominalizations (BNs) in Norwegian from a similar viewpoint. They count among 'de-verbal' nouns, but have special properties distinguishing them from the more productively derived nouns from verbs. The latter in gist follow the design below.



The next slide gives some numerical indications of how these compare to BNs.

	Verb entries in total	De-verbal nouns in total	BNs
A	500	90	27
B	800	200	72
D	460	125	30
E	260	106	20
F	1200	460	50
Total	3220	975	200

Number of de-verbal nouns and BNs based on verbs starting with *A, B, D, E, F*

These verb entries (A-F) constituting about 25% of the full verb lexicon, the proportional guess for the whole verb lexicon will be that it corresponds to about 3500 de-verbal nouns altogether, which is about 6% of all common nouns, and about 800 BNs, thus between 20 and 25% of all deverbal nouns, and 1,5% of all common nouns.

The less regular *Bare Nominalizations*

Compared to derived nouns, *Bare Nominalizations* have a much less phonologically systematic relationship to the shape of their verbal counterparts, their semantics to a much smaller extent reflects the valency and aspect of the corresponding verb, and their gender varies from BN to BN in ways non-predictable from a putative verbal source, as exemplified:

Masculine	Neuter
gang related to *gå* (‘go’)	*fall* related to *falle* (‘fall’)
søvn related to *sove* (‘sleeep’)	*løp* related to *løpe* (‘run’)
	hopp related to *hoppe* (‘jump’)

Gender of ‘bare nominalizations’

This raises the question how to lexically define BNs: they cannot have a verbal source, since gender has to be part of their lexical definition. Solution: an abstract *root* common to both verb and BN.

Illustration of BN vs. derived noun

(1) Hun var sykmeldt grunnet fall fra hest under ***løp***
She was sick-leaved because of fall from horse during run

(2) Hun gikk med krykker grunnet overanstrengelse av achillessenen under ***løping***
She was walking with crutches due to over-exertion of her achilles-tendon during running

Løping in (2) expresses the same aspectual properties as the verb does, and finds a c-commanding controller in the structure in the way a grammatically active argument does, while *løp* in (1) lacks both of these properties. Thus *løping* can be seen as having a grammatically active argument structure, but *løp* not. In the verb sign, SEM represents semantic argument structure, and so it does in the entry for *løping*:

$$\begin{bmatrix} \text{ORTH} \langle \text{"}løping\text{"} \rangle \\ \text{HEAD} [noun] \\ \text{SEM} \begin{bmatrix} \text{PRED løpe\_rel} \\ \text{ARG1} \boxed{1} \end{bmatrix} \\ \text{SIT } run \begin{bmatrix} \text{ACTOR} \boxed{1} \\ \text{MOVER} \boxed{1} \\ \text{PROTR} + \\ \text{DYN} + \end{bmatrix} \end{bmatrix}$$

'Root' representation for Bare Nominalizations

Root sign:

$$\begin{bmatrix} \text{ORTH} \langle \text{"}løp\text{"} \rangle \\ \text{SIT } run \begin{bmatrix} \text{ACTOR } \boxed{3} \\ \text{MOVER } \boxed{3} \end{bmatrix} \end{bmatrix}$$

Verb sign:

$$\begin{bmatrix} \text{ORTH} \langle \text{"}løpe\text{"} \rangle \\ \text{HEAD} [verb] \\ \text{SUBJ} \left\langle \begin{bmatrix} \text{INDX.} \boxed{1} \end{bmatrix} \right\rangle \\ \text{SEM} \begin{bmatrix} \text{PRED løpe\_}rel \\ \text{ARG1 } \boxed{1} \end{bmatrix} \\ \text{SIT } run \begin{bmatrix} \text{ACTOR } \boxed{1} \\ \text{MOVER } \boxed{1} \\ \text{PROTR } + \\ \text{DYN } + \end{bmatrix} \end{bmatrix}$$

BN sign:

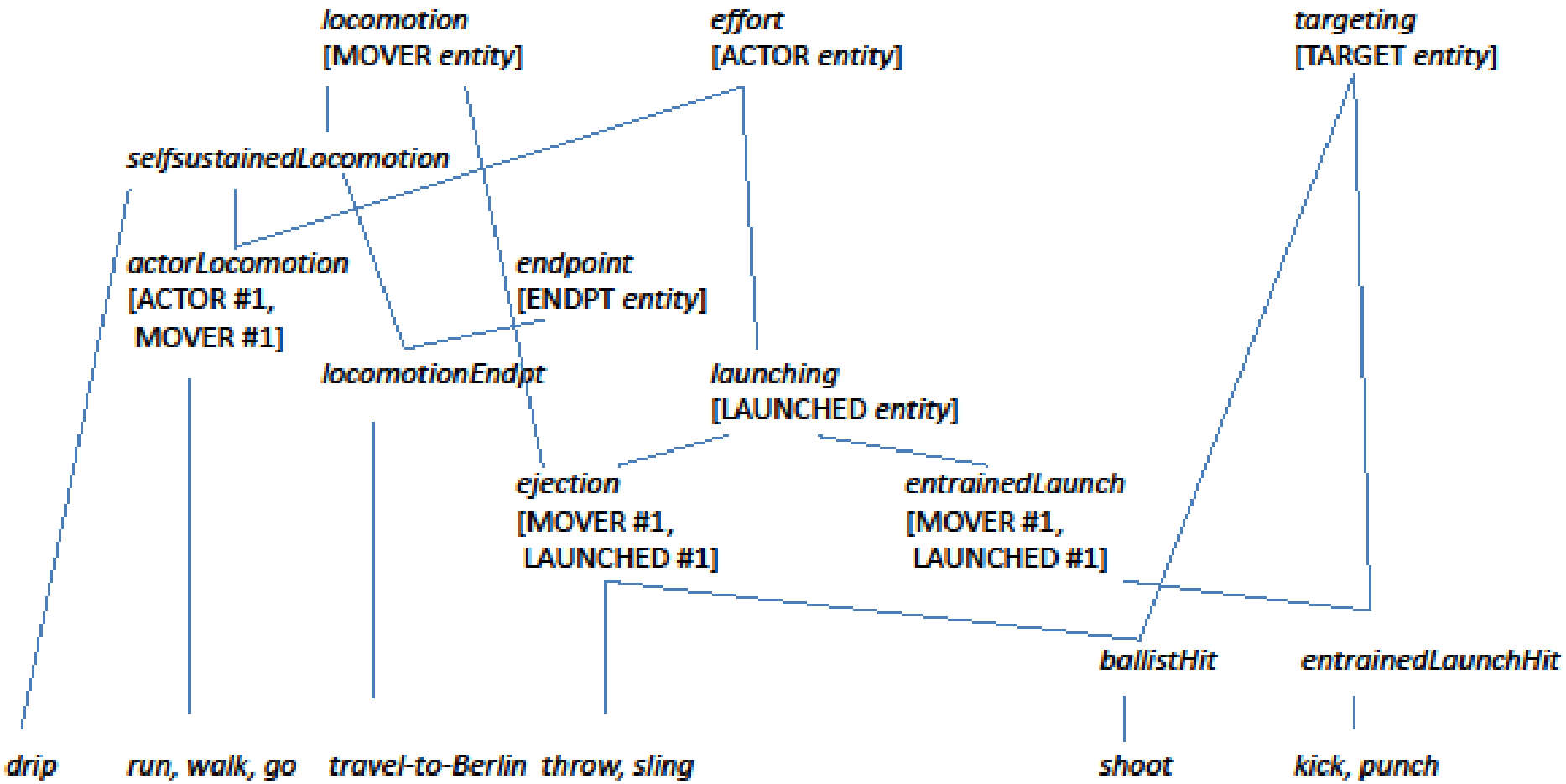
$$\begin{bmatrix} \text{ORTH} \langle \text{"}løp\text{"} \rangle \\ \text{HEAD} \left[noun \left[\text{GENDER } neut \right] \right] \\ \text{SIT } run \begin{bmatrix} \text{ACTOR } \boxed{3} \\ \text{MOVER } \boxed{3} \\ \text{PROTR } + \\ \text{DYN } + \end{bmatrix} \end{bmatrix}$$

Situational specification

The SIT attribute can be seen as opening the window to ’free’ semantics, not bound to syntactic structure, and in this respect distinct from the SEM attribute. The information encoded here can thus be quite fine-grained. The following AVM illustrates how this can be fully achieved with the TSF formalism, where *run* is here a situational type located within a large type hierarchy as indicated in the next slide, and the feature specification corresponds to expositions that could be done also in formats like tables, as illustrated in the subsequent slide.

$$
\textit{run}\left[\begin{array}{l}
\text{ACTOR } \boxed{1} \\
\\
\text{MOVER } \boxed{1}\ \textit{running}-\textit{partcpnt}\left[\begin{array}{l}
\text{ITERATN } \textit{zip}-\textit{lock} \\
\text{INCREM}-\text{DIM } \textit{horisontal} \\
\text{PARTorORGAN } \textit{leg} \\
\text{VELOCITY } \textit{fast} \\
\text{MEDIUM } \textit{terfrma}
\end{array}\right] \\
\\
\text{DYN } + \\
\text{PROTR } +
\end{array}\right]
$$

Excerpt of a possible situation-type hierarchy



Sense display (Beermann)

German	English	Norw	KIND	INCREM-DIM	ITER-PATT	VELOCITY	PART/ORGAN	MEDIUM
Laufen	*run/go*	*løpe/gå*	legged-anim	horisontal	zip-lock	unmarked	all-legs	terfrm
Gehen	*walk*	*gå*	legged-anim	horisontal	zip-lock	unmarked	all-legs	terfrm
Rennen	*Run*	*løpe*	legged-anim	horisontal	zip-lock	fast	all-legs	terfrm
Schlendern	*stroll*	*slentre*	human	horisontal	zip-lock	slow	both-legs	terfrm
Spazieren	*stroll*	*spasere*	human	horisontal	zip-lock	slow	both-legs	terfrm
Klettern	*climb*	*klatre*	legged-anim	vertical	zip-lock	unmarked	all-limbs	terfrm
Hinken		*hinke*	2- egged-anim	horisontal	succession	unmarked	one-leg	terfrm
	creep	*krype*	Anim	horisontal	succession	unmarked	under-side	terfrm
Fliegen	*fly*	*fly*		all	succession	unmarked	wings/arms	air
Schwimmen	*swim*	*svømme*	All	all	succesion	unmarked	All-limbs	water

The Situation space – and multilingual grammar

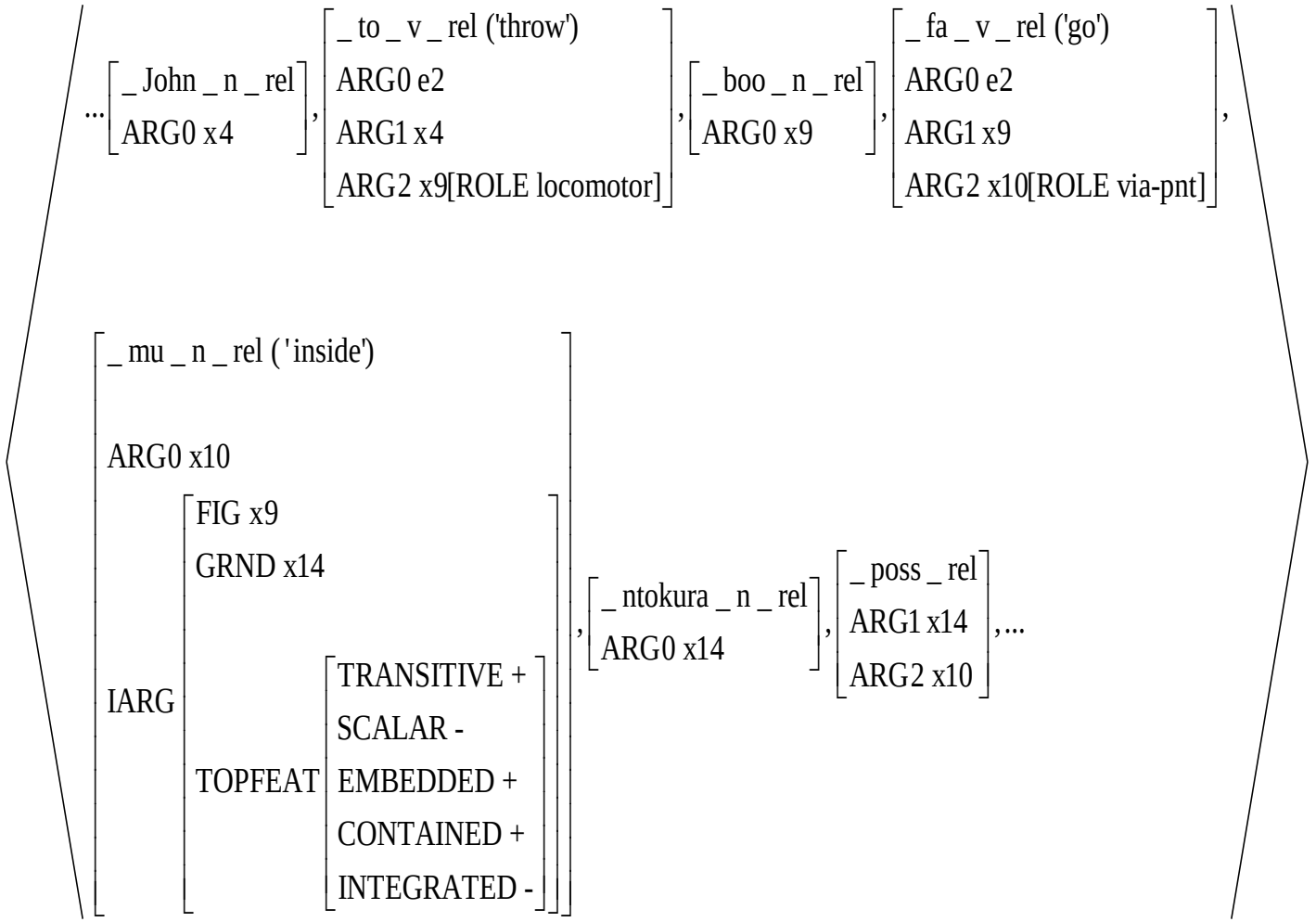
This sketch illustrates how a representation of situational specifications can be integrated in the same overall format as the rest of the information encoded in the grammar. One interesting aspect of this level of representation is that it ’ignores’ the lexical constituency of a sentence, as when verb and noun in an LVC constitute a single situational matrix. Thus, the content of one given sitution may well be expressed in more than one word. In a cross-linguistic perspective such an assumption can provide a format for analysis of supposedly mono-event Serial Verb Constructions such as (1), with tentative MRS and SIT shown on the next slide. The corresponding construction in a different language would have a different MRS, but perhaps the same SIT, opening for an *interlingua* perspective relative to translation.

(1) *John too* ɔ*bo*ɔ *no faa ntokura no mu*. (Akan)

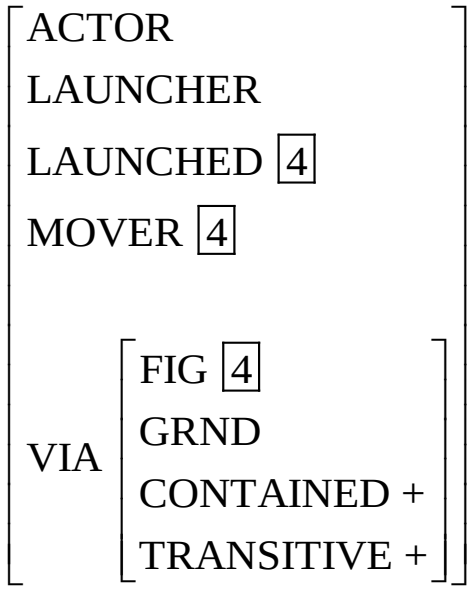
John	to-o	ɔ-boɔ	no	fa-a	ntokura	no	mu.
John	throw-PST	SG-stone	DEF	pass-PST	window	DEF	inside
N	V	N	DET	V	N	DET	N

'John threw the stone through the window’

MRS (= 'SEM')


... _ John _ n _ rel
ARG0 x4
_ to _ v _ rel ('throw')
ARG0 e2
ARG1 x4
ARG2 x9[ROLE locomotor]
_ boo _ n _ rel
ARG0 x9
_ fa _ v _ rel ('go')
ARG0 e2
ARG1 x9
ARG2 x10[ROLE via-pnt]
_ mu _ n _ rel ('inside')
ARG0 x10
IARG
FIG x9
GRND x14
TOPFEAT
TRANSITIVE +
SCALAR -
EMBEDDED +
CONTAINED +
INTEGRATED -
_ ntokura _ n _ rel
ARG0 x14
_ poss _ rel
ARG1 x14
ARG2 x10
...

SIT


ACTOR
LAUNCHER
LAUNCHED 4
MOVER 4
VIA
FIG 4
GRND
CONTAINED +
TRANSITIVE +

Tying up this far

In the perspective of constructing a grammar, the specifications under SIT represent a long shot, and in a strategic development will come ’after’ the development at SEM level, still it is clear that such modules can be added incrementally to the overall grammar. Thus these demonstrations of how ’fine-grained’ aspects of analysis can be integrated in the overall structure are good demonstrations of the adequacy of the LSDG-sustaining formalism itself.

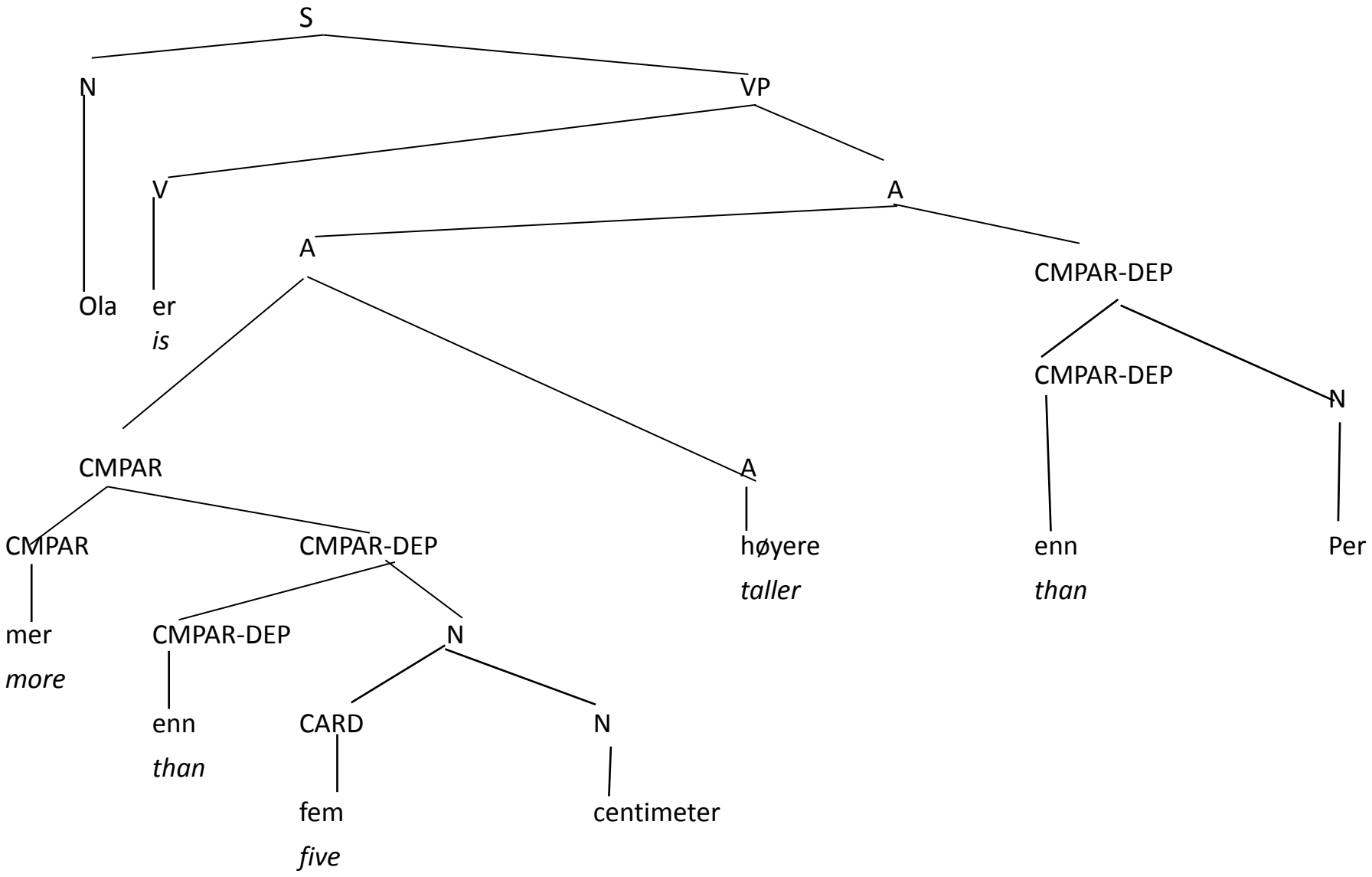
On this note, we quickly show some further examples of the integration of state-of-the-art analyses in the formalism, in these cases fully implemented in NorSource.

Further analyses provided by the parser

We first illustrate how arithmetic content as expressed in a normal sentence can receive a representation in what can safely be called a *logical form* (based on Davis and Hellan 1975), on which one could envisage that some kind of logical operations could be performed. The syntactic part of the parser construes the input sentence with a tree simplified shown as in the next slide, and the Minimal Recursion Semantics (MRS) assigned to it is shown on the subsequent slide (from the Norsource web parser 11.08.2017).

In principle this is also a type of information which could be included in a large corpus produced from the grammar. However, a ’down-stream’ converter of the MRS into a logical reasoner would also be conceivable.

The second type of analytic output, also carried by the MRS (from the Norsource web parser 11.08.2017), exposes detailed spatio-temporal information (following Trujillo 1996, Beermann and Hellan 2005, 2006), again induced by the grammar and a candidate for another large scale inclusion in a corpus, and this time also an imaginable candidate for ’down-stream’ employment for instance in robot instructions related to spatial movement.


S
N
VP
V
A
A
CMPAR-DEP
Ola
er
is
CMPAR-DEP
N
CMPAR
A
CMPAR
CMPAR-DEP
høyere
taller
enn
than
Per
mer
more
CMPAR-DEP
N
enn
than
CARD
N
fem
five
centimeter

MRS for ”Ola er mer enn fem centimeter høyere enn Per”

ltop=h0, index=e1
h3:named_rel([carg:***ola, arg0:x2***])
h5:_def_q_rel([arg0:x2, rstr:h6, body:h7])
h8:_være-property_rel([arg0:e1, arg1:h9])
h9:***_mer-exceed_c_rel([arg0:u10, arg1:u11, arg2:x12***])
h9:_measure_out_rel([arg0:u10, arg1:u11, arg2:u13])
h14:_cmparpartcl_rel([arg0:u15])
h16:_card_rel([carg:5-rel, arg0:u18, arg1:x12])
h16:***_centimeter_n_rel([arg0:x12***])
h19:_plurindef_q_rel([arg0:x12, rstr:h20, body:h21])
h22:***_høy_a_rel([arg0:u23, arg1:x2***])
h22:***_exceed_c_rel([arg0:u13, arg1:u23, arg2:u24***])
h9:***_compare_rel([arg0:u24, arg1:x25***])
h26:named_rel([carg:***per, arg0:x25***])
h28:_def_q_rel([arg0:x25, rstr:h29, body:h30])
< qeq(h6,h3), qeq(h20,h16), qeq(h29,h26) >

An interesting point about the derivation of this ’logical form’ is that it must be done directly off the syntactic tree – no ’intermediate’ word-by-word MRS would allow for this more detailed form to be constructed.

MRS for 'Han springer til skogen på fem minutter' (*He runs to the forest in five minutes)*

ltop=h0, index=e1
h3:named_rel([carg:ola, arg0:x2])
h5:_def_q_rel([arg0:x2, rstr:h6, body:h7])
h8:_springe_v-intr_rel([arg0:e1, arg1:x2])
h8:_til_p-dirtel_rel([arg0:u9, arg1:x2, arg2:x10, iarg:u11])
h12:_skog_n_rel([arg0:x10])
h13:_def_q_rel([arg0:x10, rstr:h14, body:h15])
h8:_på_p-temp_rel([arg0:u16, arg1:e1, arg2:x17, iarg:u18])
h19:_card_rel([carg:5-rel, arg0:u21, arg1:x17])
h19:_minutt_n_rel([arg0:x17])
h22:_plurindef_q_rel([arg0:x17, rstr:h23, body:h24])
< qeq(h6,h3), qeq(h14,h12), qeq(h23,h19) >
e1, sort=verb-act-specification, sf=prop, e.tense=pres, e.mood=indicative
x2, wh=-, png.ng.num=sing, png.ng.gen=m, png.pers=thirdpers, role=***mileage-obj***
u9, sort=verb-act-specification
x10, wh=-, bounded=+, sort=inan-***endpnt-of-path***, png.ng.num=sing, png.ng.gen=m, png.pers=thirdpers, ***role=endpnt***
x17, wh=-, bounded=+, png.ng.num=plur, png.ng.gen=n, png.pers=thirdpers
u18, class.***co-extensive-with=+***
u21, png.ng.num=plur, png.ng.gen=n, png.pers=thirdpers

Constructing an e-learning tool from an LKB grammar

For the record, NorSource has one quite active 'down stream' application, namely The *Norwegian Online Grammar Sparrer* ***(***Hellan et al. 2013), an online language training tool, with a direct access point at

http://regdili.hf.ntnu.no:8081/studentAce/parse and a wiki access point at

http://typecraft.org/tc2wiki/A_Norwegian_Grammar_Sparrer

It accepts *batches of up to 10 sentences*, each with max. 10 words. ***Responses*** are given (in order of creation) in ***English, Polish, Italian, German, Chinese, Bulgarian, Arabic*** , and ***Norwegian,*** accompanied by a facility for ***generating*** the correct alternative, for direct illustration. Pages with examples of feedback messages are also available on TypeCraft for most languages, while accompanying wiki pages describing aspects of Norwegian grammar – now about 20 pages – are in English.

Here it is the processor itself which carries the functionality, however it would probably not be possible to sustain such functionalities – especially the *generation* function to be illustrated – without a facility like the MRS.

Illustration of responses and generation are given below.

The Sparrer – illustrating a small batch, responses in English

Norwegian Grammar Tutor

Demo with ACE, version 2.2 for further guidelines, see Info

Enter up to 10 sentences with up to 10 words each and press the Analyze button.
Always use a verb and a subject in your sentences

Feedback Language English

```
mannen smile
et gutt ikke kommer
jeg ser gutten sin
Ola skammer
```

Analyze

generate	The word "smile" is in infinitive, but should be in past or present tense. More description
generate	A determiner must have the same gender, number and definiteness as the noun it modifies. More description An adverb cannot precede the finite verb in a main clause.
generate	The reflexive pronoun "sin" does not match the person of the word it refers back to. Try using "min" instead. More description
generate	The verb "skammer" requires a reflexive object. More description

Generation for sentence 2 – “*et gutt ikke kommer”

Norwegian Grammar Tutor

Demo with ACE, version 2.2 for further guidelines, see Info

Enter up to 10 sentences with up to 10 words each and press the Analyze button.
Always use a verb and a subject in your sentences

Feedback Language italiano

```
mannen smile
et gutt ikke kommer
jeg ser gutten sin
Ola skammer
```

Analyze

Grammar Option(s) for Sentence

#	Sentence
1	En gutt kommer ikke

Generation for sentence 3 – “*Jeg ser gutten sin”

Norwegian Grammar Tutor

Demo with ACE, version 2.2 for further guidelines, see Info

Enter up to 10 sentences with up to 10 words each and press the Analyze button.
Always use a verb and a subject in your sentences

Feedback Language italiano

```
mannen smile
et gutt ikke kommer
jeg ser gutten sin
Ola skammer
```

Analyze

Grammar Option(s) for Sentence

#	Sentence
1	Jeg ser gutten min

Controllo della grammatica norvegese

Incatenata a Messaggi di feedback.

Questo sistema è additato a procurare feedback grammatico alle frasi che hai scelto.

Per favore, cliccare sull'immagine del troll  **per trovare il controllo grammatico.**

Nella finestra che appare, scrivere una o più frasi (fino a dieci frasi alla volta, cambiando riga dopo ogni frase, e ogni frase costituendo non più di 10 parole) e premere il pulsante 'Analizza'. Se la frase è grammaticalmente ben formata, avrai la risposta:

```
Una frase norvegese ben formata.
```

Se la frase non è gramaticalmente ben formata, ci sono due possibilità:

Diagnosi dell'errore

La prima possibilità è che il sistema ti dice che cosa nella frase è sbagliato. Ad esempio, alla frase sgrammaticata

```
Jeg liker du.
```

dove la forma di soggetto "du" è usata come oggetto, avrai la risposta:

```
La parola "du" è marcata con il caso sbagliato, prova invece a usare "deg".
```

Insieme con un tale messaggio di errore, può apparire un pulsante 'Generate'.Se lo clicchi avrai una nuova finestra mostrando una versione raccomandata della frase desiderata, in questo caso:

```
Jeg liker deg.
```

Conclusion

As remarked at the outset, the perhaps most interesting linguistic aspect of LSDGs is their capacity as ’complete functions’ from data to analysis, and thus as symbolic models of ’total grammars’ as such. That is not an easy focus to maintain, since what are we then comparing our object of investigation with? In the deep sea fish metaphor, what is the ’water’ – the total symbolic system with its conditioning effects on the individual pieces of ’symbolic tissue’, or the embeddedness of this system in the wider ecology of co-understanding which constitutes communication? Let’s say either, although our concern here has been the former, with mentioning of large structures, whose content-by-their-size we are perhaps merely beginning to ask questions about; but we are at least able to articulate the more individual pieces (analytically). Describing them doesn’t necessarily have to be in the medium of an LSDG, but what is good about the LSDGs is that your descriptions are operationally put to test every time you parse a text, and that in order to compose a description which can make it to the parsing point at all (with all the content in question), you are already building/subscribing to/using an explicit overall model; thus fighting the elements directly, rather than selectively writing about them.

References

- Beermann, Dorothee and Mihaylov, Pavel, 2014. Collaborative databasing and Resource sharing for Linguists. *Languages Resources and Evaluation* 48. Dordrecht: Springer, 1-23.
- Copestake, A. (2002). *Implementing Typed Feature Structure Grammars*. CSLI Publications.
- Copestake, A., D. Flickinger, I. Sag and C. Pollard. (2005). Minimal Recursion Semantics: an Introduction. *Journal of Research on Language and Computation*. 281-332.
- Dakubu, M.E. Kropp and Lars Hellan, 2017. A labeling system for valency: linguistic coverage and applications. In Hellan, L., Malchukov, A., and Cennamo, M (eds) *Contrastive studies in Valency*. Amsterdam & Philadelphia: John Benjamins Publishing Co.
- Davis, Anthony. 2000. *The Hierarchical Lexicon*. Stanford; CSLI Publications.
- Goldhahn, D., Eckart, T., Quasthoff, U. (2012). Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), 2012.
- Grimshaw, Jane, and Armin Mester. (1988). Light Verbs and Θ-Marking. *Linguistic Inquiry* 19(2):205–232.
- Grimshaw, Jane. 2005[1993]. Semantic structure and semantic content in lexical representation. In *Words and Structure*, Jane Grimshaw, 75–89. Stanford CA: CSLI.
- Heift, T., and M. Schulze. (2007). *Errors and Intelligence in Computer-Assisted Language Learning: Parsers and Pedagogues*. Routledge, New York.
- Hellan, L. (2008) From Grammar-Independent Construction Enumeration to Lexical Types in Computational Grammars. COLING (http://www.aclweb.org/anthology-new/W/W08/#1700).
- Hellan, Lars. 2016. Light Verb Constructions as valency modeling. A study of Norwegian. Presented at SLE 2016, Naples. Submitted for proceedings as: Unification and selection in Light Verb Constructions. A study of Norwegian.
- Hellan, Lars.. 2017. A design for the analysis of bare nominalizations in Norwegian. In: A. Malicka-Kleparska & M. Bloch-Trojnar (eds) *Aspect and Valency in nominals*. Berlin: Mouton de Gruyter.
- Hellan, Lars. Forthcoming. Situations in Grammar. In Bodomo, A., Essegbey, J. and Kallulli, D. (eds). Title and publisher to be announced.

References

- Hellan, L. and D. Beermann. 2005. Classification of Prepositional Senses for Deep Grammar Applications. In: Kordoni, V. and A. Villavicencio (eds.) *Proceedings of the 2nd ACL-Sigsem Workshop on The Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications,* Colchester, UK.
- Hellan, Lars and M.E. Kropp Dakubu, 2010*. Identifying verb constructions cross-linguistically*. In *Studies in the Languages of the Volta Basin* 6.3. Legon: Linguistics Department, University of Ghana.
- Hellan, L. and Beermann, D. (2014) Inducing grammars from IGT. In Z. Vetulani and J. Mariani (eds.) *Human Language Technologies as a Challenge for Computer Science and Linguistics*. Springer.
- Hellan, L., D. Beermann, T. Bruland, M.E.K. Dakubu, and M. Marimon (2014) *MultiVal*: Towards a multilingual valence lexicon**.** In Calzolari et al. (eds) 2014.
- Hellan, L., Bruland, T., Aamot, E., Sandøy, M.H. (2013): A Grammar Sparrer for Norwegian. *Proceedings of NoDaLiDa* 2013.
- Hellan, Lars and Tore Bruland. 2015. A cluster of applications around a Deep Grammar. In: Vetulani et al. (eds) *Proceedings from The Language & Technology Conference* (LTC) 2015, Poznan.
- Adam Przepiórkowski, El´zbieta Hajnicz, Agnieszka Patejuk, Marcin Woli´nski, Filip Skwarski, and Marek ´Swidzi´nski. (2014). Walenty: Towards a comprehensive valence dictionary of Polish. In Calzolari et al. (eds) 2014.
- Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014), pages 2785–2792, Reykjavík, Iceland. ELRA.
- Osenova, Petya (2011). Localizing a Core HPSG-based Grammar for Bulgarian. In: Hanna Hedeland, Thomas Schmidt, Kai Worner (eds.) *Multilingual Resources and Multilingual Applications*, *Proceedings of GSCL 2011*, ISSN 0176-599X, Hamburg, pp. 175-180.
- Patejuk, Agnieszka (2016). Integrating a rich external valency dictionary with an implemented XLE/LFG grammar. In Doug Arnold, Miriam Butt, Berthold Crysmann, Tracy Holloway King, Stefan Müller (editors) *Proceedings of the Joint 2016 Conference on Head-driven Phrase Structure Grammar and Lexical Functional Grammar.* Stanford: CSLI Publications, pp 520-540.
- Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar.* Chicago University Press.
- Pompei, A. and V. Piunno (2015). Light Verb Constructions. An interlinguistic analysis to explain systemic irregularities. Paper presented at SLE 2015, Leiden.