Proceedings of the 16th Joint ACL-ISO Workshop Interoperable Semantic Annotation (ISA-16), pages 22–31
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Interoperable Semantic Annotation

Lars Hellan
Norwegian University of Science and Technology, Norway
lars.hellan@ntnu.no

Abstract

The paper presents an annotation schema with the following characteristics: it is formally compact; it systematically and compositionally expands into fullfledged analytic representations, exploiting simple algorithms of typed feature structures; its representation of various dimensions of semantic content is systematically integrated with morpho-syntactic and lexical representation; it is integrated with a 'deep' parsing grammar. Its compactness allows for efficient handling of large amounts of structures and data, and it is interoperable in covering multiple aspects of grammar and meaning. The code and its analytic expansions represent a cross-linguistically wide range of phenomena of languages and language structures. This paper presents its syntactic-semantic interoperability first from a theoretical point of view and then as applied in linguistic description.

Keywords: semantic annotation tags, typed feature structures, valence, semantic argument structure, situation structure, quantifier scope, Ga , Norwegian

1. Introduction[1]

Semantic annotation can cover, amongst others, semantic argument structure; situation structure; quantifier scope; perspective of wording (transparent vs. oblique); anaphora; turns in discourse and types of moves or states within larger texts. Semantic annotation necessarily applies to linguistic expressions or texts, and the assigned content is often dependent on grammatical or lexical analysis, calling for *grammatically/lexically interoperable* annotation designs. This means that a natural format for semantic annotation is one where it interacts with grammatical or lexical representation more generally. In most areas the degree of complexity of the semantic representation, combined with the complexity of lexical or grammatical specification of the phenomena to which it is applied, is so high that it is reasonable to use a system of compact semantic *tagging*.

We here present a system of integrated morpho-syntactic and semantic tagging applicable to large constructs such as verb valence lexicons and corpora tagged for valence. The tagging system we present is an extension of the system Construction Labeling (CL) described and applied in Hellan and Dakubu 2010 and Dakubu and Hellan 2017. In this extended system, the CL code is mapped to a a Typed Feature Structure (TFS) formalism sustaining computational 'deep' parsers assigning both morphosyntactic and semantic analysis to the sentences parsed. The formalism of the system comes close to the HPSG formalism,[2] but with important exceptions (see below), and alternatives can be explored relative to other frameworks as well, such as, in all likelihood, LFG.[3]

The first part of the paper is devoted to the overall formal architecture of the system, in particular presenting its semantic components both inside of the TFS system and in the tagging formalism (sections 2-4). In the second part (section 5) we describe how the overall tagging formalism can be employed in semantic specification in large resources such as valence lexicons and valence corpora, first addressing a valence lexicon and corpus for the West African language Ga (Kwa, spoken in Ghana), and then valence resources for Norwegian. In the third part (section 6) we mention possible extensions of the system from the argument structure domain to quantifier scope and other scopal phenomena.

2. Annotation related to semantic argument structure of verbal constructions

The *Construction Labelling (CL)* code provides construction-level annotation tags which in one-line strings provide much of the information that could otherwise be expressed in multi-tier syntactic and semantic annotation. The strings are subject to semi-automatic consistency control, and can also be applied in valence specification in lexicons and in grammatical parsing. It has the added capacity of serving as types in a TFS system, enabling the consistency control and the parsing functionality. Following the overall left-to-right order indicated in (1), CL valency annotations are written as illustrated in (2):

(1)

head – valenceFrame – special properties of syntactic constituents – semantic roles of constituents – aspect, Aktionsart – situation type

(2) *v-tr-suAg_obAffincrem-ACCOMPL*
[Ex. John ate the cake]

The string in (2) reads: 'a verb-headed transitive syntactic frame where the subject carries an agent role and the object an incrementally affected role, and the situation type expressed is 'accomplishment'.

The example (3) from Citumbuka (Bantu) instantiates verbal derivation underlying the expression of *causation*, illustrating interplay between morpho-syntax and semantics:

[1] I am grateful to the three reviewers for their helpful comments.

[2] On HPSG ('Head-Driven Phrase Structure Grammar), see Pollard and Sag 1994 and Copestake 2002.

[3] On LFG ('Lexical Functional Grammar'), see Bresnan (2001).

(3)

Mary	wa-ka-mu-phik-**isk**-a	John	nchunga
Mary	1SM-Pst-1OM-cook-Caus-FV	John	beans
N	V	N	N
	vCaus-dbobCs		

'Mary made John cook beans'

The CL-string *vCaus* means that the head is a verb and has a causative morpheme, and *dbobCs* means that the construction is a double object construction 'derived' through causativization. The derivation can be further indicated through specifications of the arguments of the derived verb, in terms of their derivational histories, thus extending the formula *vCaus-dbobCs* to

(4) *vCaus-dbobCs-suC-obCsu-ob2Cob*

where the added items read as follows:

suC - *subject is created by Causativization*

obCsu - *object is derived ('demoted') from subject by Causativization*

ob2Cob - *second object is derived from object by Causativization*

Expanding from what was said above (cf. (1)), each CL tag is a string consisting of, first, a label specifying POS of head of the construction and salient morphological marking (like *vCaus* in (3)), second, a label designating the overall structure of a construction (encoding notions like intransitive, transitive, ditransitive/double object, etc. (such as *dbobCs* in (3)), third a string of labels classifying features of the arguments such as the added tags above - first syntactic features and then semantic features -, and finally a string of labels for TAM features and situational content.

Whenever a putative CL string is composed, the labels of the string have to match with each other – for instance, if one label is *intr*, for 'intransitive', then there cannot be an argument label prefixed by *ob*, since *intr* is not defined for such a label. A processing mechanism enforcing such consistency is provided using a unification-based TFS system, in which the CL tag labels are defined as *types*. This TFS system is at the same time what underlies the automatic interoperability.

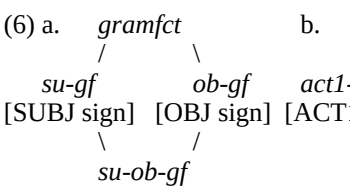
Information in such a system is generally exposed through Attribute Value Matrices (AVMs), where each AVM belongs to a type, and attributes are introduced (declared) according to the following conventions:

(5) *[A]* A given type introduces the same attribute(s) no matter in which environment it is used.

[B] A given attribute is declared by one type only (but occurs with all of its subtypes).

Among the types in the present system are types for *grammatical functions* represented as values of the attribute '*GF*' (which constitutes an addition to the standard array of formal notions in HPSG) [4], and roles in *semantic argument structure* represented as values of the attribute 'ACTNT' (see further in section 4 for types for *situation structure*). The type *gramfct* has subtypes declaring GF attributes such as 'SUBJ' and 'OBJ' (cf. (6a)), and the type *actnt* has subtypes declaring semantic participant attributes, such as 'ACT1' and 'ACT2' (cf. (6b)):

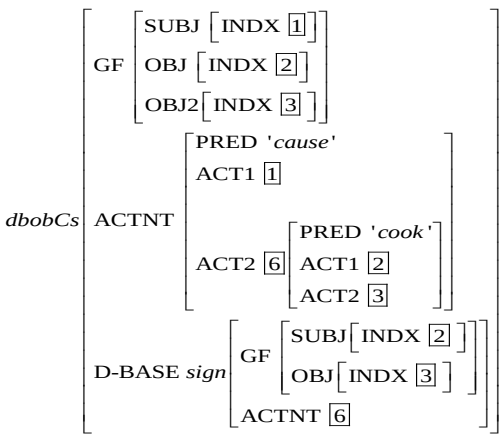
(6) a.



b.

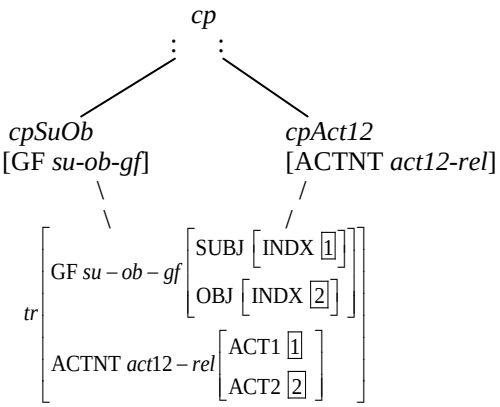
With such features as basis, one can represent, e.g., (3) as in (7), which is an AVM representing a *construction*. This involves a specification of grammatical functions and actants acting together, identified through the attributes GF and ACTNT, introduced at the level of constructions through a declaration '*cp* := *top* & [GF *gramfct*, ACTNT *actnt*]', where the type label *cp* stands for 'construction profile'. It has *dbobCs* as one of its subtypes, comprising the notion of *causation* through the PRED value 'cause' inside the ACTNT specification; the attribute 'D-BASE' stands for 'derivational base' (or 'input'):

(7) AVM for double object construction with causative semantics and causative derivation (cf. (3)):



We then outline how, in their capacity as types in the TFS grammar, CL labels define AVMs at the formal level of *constructions*. As subtypes of *cp*, definitions sustaining the type *tr* ('transitive') are shown in the following, achieved through a join of two *cp* subtypes, one defined with regard to GF, the other with regard to ACTNT:

(8)



[4] For discussion and motivation for the emplyment of GF notions in an HPSG-based formalism, see Hellan (2019a) and Hellan (2020).

tr is thus formally defined as a type of *construction*. Similar depths of specification are required for all CL labels.

When CL labels occur in a string, they *unify*. To illustrate with some types relevant also for English, the types to which the labels in (2) correspond are indicated in (9), and their unification is (10):

(9)

v - - - [HEAD *verb*]

tr - - - (cf. (8))

suAg - - - $\left[\left[\text{GF}\left[\text{SUBJ}\left[\text{INDX}\ \boxed{1}\right]\right]\right]\ \left[\text{SIT}\left[\text{ACTOR}\ \boxed{1}\right]\right]\right]$

obAffincrem - - - $\left[\text{GF}\left[\text{OBJ}\left[\text{INDX}\ \boxed{1}\right]\right]\right]\ \left[\text{SIT}\left[\text{AFFECTED}\ \boxed{1}\right]\right]$

ACCOMPL - - - [SIT *accomplishment*]

(10)

$$\left[\begin{array}{l}\text{HEAD } \textit{verb}\\ \text{GF}\left[\begin{array}{l}\text{SUBJ}\left[\text{INDX}\ \boxed{1}\right]\\ \text{OBJ}\left[\text{INDX}\ \boxed{2}\right]\end{array}\right]\\ \text{ACTNT}\left[\begin{array}{l}\text{ACT1}\ \boxed{1}\\ \text{ACT2}\ \boxed{2}\end{array}\right]\\ \text{SIT } \textit{accomplishment}\left[\begin{array}{l}\text{ACTOR}\ \boxed{1}\\ \text{AFFECTED}\ \boxed{2}\end{array}\right]\end{array}\right]$$

The semantic roles corresponding to the labels *suAg* and *obAffincrem* are represented in a space of semantics called *Situation Structure* ('SIT') through the attributes ACTOR and AFFECTED, both relevant within the situation type *accomplishment*; cf. section 4 below.

Returning to the label in (4), the AVMs for the 'derivational histories' will be as in (11), the unification with the structure for *dbobCs* is the structure in (7).

(11)

a. $\textit{suC}\left[\begin{array}{l}\text{GF}\left[\text{SUBJ } \textit{sign}\left[\text{INDX}\ \boxed{1}\right]\right]\\ \text{ACTNT}\left[\begin{array}{l}\text{PRED } \textit{'cause'}\\ \text{ACT1}\ \boxed{1}\end{array}\right]\end{array}\right]$

b. $\textit{obCsu}\left[\begin{array}{l}\text{GF}\left[\text{OBJ}\left[\text{INDX}\ \boxed{2}\right]\right]\\ \text{ACTNT}\left[\begin{array}{l}\text{PRED } \textit{'cause'}\\ \text{ACT2}\ \boxed{6}\left[\text{ACT1}\ \boxed{2}\right]\end{array}\right]\\ \text{D-BASE } \textit{sign}\left[\begin{array}{l}\text{GF}\left[\text{SUBJ}\left[\text{INDX}\ \boxed{2}\right]\right]\\ \text{ACTNT}\ \boxed{6}\end{array}\right]\end{array}\right]$

c. $\textit{ob2Cob}\left[\begin{array}{l}\text{GF}\left[\text{OBJ2}\left[\text{INDX}\ \boxed{3}\right]\right]\\ \text{ACTNT}\left[\begin{array}{l}\text{PRED } \textit{'cause'}\\ \text{ACT2}\ \boxed{6}\left[\text{ACT2}\ \boxed{3}\right]\end{array}\right]\\ \text{D-BASE } \textit{sign}\left[\begin{array}{l}\text{GF}\left[\text{OBJ}\left[\text{INDX}\ \boxed{3}\right]\right]\\ \text{ACTNT}\ \boxed{6}\end{array}\right]\end{array}\right]$

Here each type represents the part of the whole AVM corresponding to the content of the (derived) subject, object, and second object ('ob2'). Unification presupposing feature compatibility, being the formal point illustrated here, the control of consistency in the CL string (4) is thereby inbuilt in the formalism. We have at the same time introduced two aspects of semantic analysis represented by the attributes ACTNT and SIT, to which we turn further below. First we consider semantic relations carried by structures internal to NPs.

3. Annotation for semantic relations of nominal constructions

The sentence in (12) is a construction from Ga:[5]

(12)

Mi-yitso	mii-gba	mi
1S.POSS-head	PROG-split	1S
N	V	PN

"My head is aching." (literally: 'my head aches me')

Here we want to represent the subject as a possessive phrase, where the referent of the whole phrase is a (body)part of the specifier 'mi', and this specifier is also identical to the object; in terms of semantics, the situation as a whole has the label 'EXPERIENCE', the role of the subject is that of 'locus' of the experience, and the 'experiencer' is expressed by the object. In terms of the CL formalism this can be stated as follows:

(13)

v-tr-suPossp_suBPsuSpec_suSpecIDob-suLocus_obExp-EXPERIENCE

The part *suBPsuSpec* is a type representable as (14), where 'is-bodypart-of-rel' spells out 'BP', and the part *suSpecIDob* is spelled out as (15), where identical indices reflect the part 'ID':

(14)

$$\textit{suBPsuSpec}\left[\text{GF}\left[\text{SUBJ}\left[\begin{array}{l}\text{INDX}\ \boxed{1}\\ \text{GF}\left[\text{SPEC}\left[\text{INDX}\ \boxed{2}\right]\right]\\ \text{ACTNT}\left[\begin{array}{l}\text{PRED 'is-bodypart-of'-rel}\\ \text{ACT1}\ \boxed{1}\\ \text{ACT2}\ \boxed{2}\end{array}\right]\end{array}\right]\right]\right]$$

(15)

$$\textit{suSpecIDob}\left[\text{GF}\left[\begin{array}{l}\text{SUBJ}\left[\text{GF}\left[\text{SPEC}\left[\text{INDX}\ \boxed{2}\right]\right]\right]\\ \text{OBJ}\left[\text{INDX}\ \boxed{2}\right]\end{array}\right]\right]$$

Unification of (14) and (15) yields (16), adding the eventual contribution from the meaning specification of

[5] From Dakubu (Unpublished a).

the verb, and the semantic CL specifications, where 'locus' has a meaning close to 'stimulus' but in addition indicates the location of the 'stimulus':

(16)

$$\begin{bmatrix} \text{GF} & \begin{bmatrix} \text{SUBJ} & \begin{bmatrix} \text{INDX } \boxed{1} \\ \text{GF}\left[\text{SPEC}\left[\text{INDX } \boxed{2}\right]\right] \\ \text{ACTNT} & \begin{bmatrix} \text{PRED 'is-bodypart-of'-rel} \\ \text{ACT1 } \boxed{1} \\ \text{ACT2 } \boxed{2} \end{bmatrix} \end{bmatrix} \\ \text{OBJ } [\text{INDX 2}] \end{bmatrix} \\ \text{ACTNT} & \begin{bmatrix} \text{PRED 'gba'-rel} \\ \text{ACT1 } \boxed{1} \\ \text{ACT2 } \boxed{2} \end{bmatrix} \\ \text{SIT} \quad \text{exp}\textit{erience} & \begin{bmatrix} \text{LOCUS } \boxed{1} \\ \text{EXPERIENCER } \boxed{2} \end{bmatrix} \end{bmatrix}$$

We now comment on 'Situation structure' as a semantic concept, and compare it with the attribute 'ACTNT'..

4. Situation Structure

A more detailed SIT-representation of a verb like *run* is given in (17). Here *run* is a situation type.

(17)

$$\textit{run}\begin{bmatrix} \text{ACTOR } \boxed{1} \\ \text{MOVER } \boxed{1}\ \textit{running – partcpnt} & \begin{bmatrix} \text{ITERATN } \textit{zip – lock} \\ \text{INCREM – DIM } \textit{horisontal} \\ \text{PARTorORGAN } \textit{leg} \\ \text{VELOCITY } \textit{fast} \\ \text{MEDIUM } \textit{ground} \end{bmatrix} \\ \text{DYN +} \\ \text{PROTR +} \end{bmatrix}$$

Types like these sit in an ontology of situational content as outlined in Hellan (2019a, b), where each label corresponds to a node in a situation type hierarchy. Figure 1 illustrates part of such a hierarchy (also obeying the conventions in (5)).

This hierarchy hosts both general situation types and types sorting under the notions 'Aspect' or 'Aktionsart', as developed in, e.g., Vendler 1967, Smith 1991,1997, Verkuyl 1996, and many others. Attributes declared by its types can have either '+/-' as their value, or types defined within another hierarchy, instantiated by *running-participant* in (17), whose attributes represent aspects of the behavior of a participant filling the outer attribute in question (such as 'MOVER' in (17)). *Run*, together with *walk*, count as subtypes of the type *actorLocomotion*, which, in joins with general types for reaching endpoints or going by via-points, also dominates situation types for 'running to' a certain point or 'running via' a certain point, as indicated in the figure.

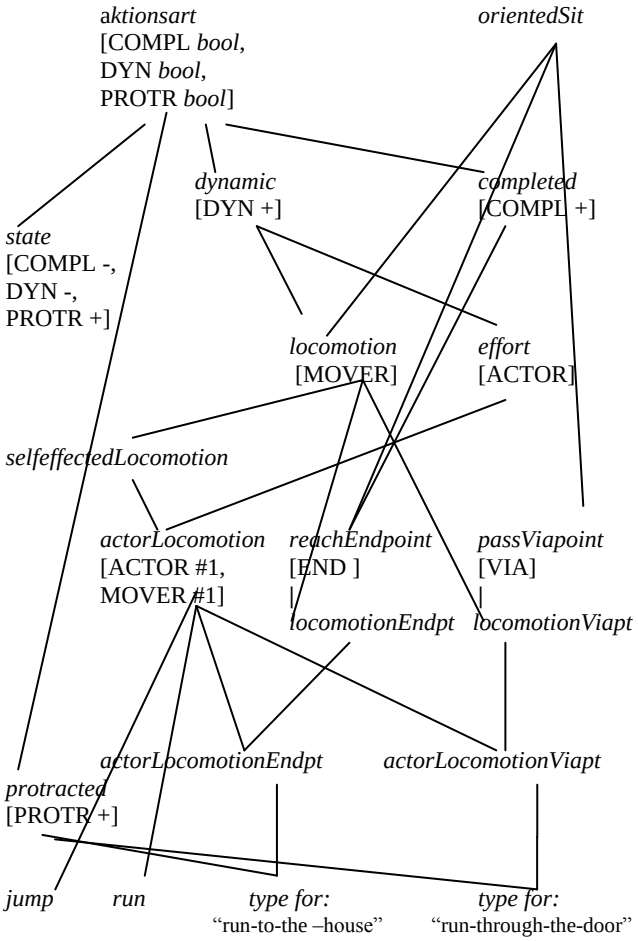


Figure 1. Partial Situation Type hierarchy

As an inheritance hierarchy, an attribute introduced for a given type will belong to all of its subtypes. Top types often introduce attributes with a value still unspecified, such as aktionsart, but once a value is set, that value holds for all the subtypes. In this way, for instance, the type *actorLocomotion* has the full structure (18), with the inherited values.

(18)

$$\textit{actorLocomotionEndpt}\begin{bmatrix} \text{ACTOR } \boxed{1} \\ \text{MOVER } \boxed{1} \\ \text{END} \\ \text{DYN +} \\ \text{COMPL +} \end{bmatrix}$$

A CL label can in principle relate to any node at any 'height' in such a hierarchy. For instance, the type *actorLocomotion* in Figure 1 corresponds to the CL label *MOTIONDIRECTED*, and its subtype *actorLocomotionEndpt* corresponds to MOTIONtoENDPT; thus, the Situation Structure assigned to a sentence like *They run to the town* will be (18), and a CL representation for the construction as such will be (19).

(19) v-intr-suDir-suEndpt-MOTIONtoENDPT

Labels such as *suAg* and *obExp* also refer into the SIT hierarchy, then into situation types sharing the attribute in question. Thus, when for instance *suMover* and

MOTIONDIRECTED occur in the same CL string, then the attribute MOVER is declared by a type also dominating *actorLocomotion*.

The expressiveness of this format of representation may be compared with formats of representation in Lexical Semantics using predicate-argument style notation, for instance considering the notation for lexical semantics developed in Jackendoff 1990, called '*Lexical Conceptual Structure' (LCS), (*see also Dorr 1993). Here a sentence like *John ran into the room* will have the representation (20), where the predicate 'GO' represents dynamic directional movement:

(20)
$[GO\ ([JOHN]_{THING}, [TO\ ([IN\ ([ROOM]_{THING})]_{PLACE})]_{PATH})]_{EVENT}$

Relative to this LCS formula, the type *actorLocomotion* corresponds to 'GO', and *reachEndpoint* corresponds to the predicate 'TO'. The dominance by *dynamic* corresponds to the bracket label 'EVENT', and *orientedSit* corresponds to 'PATH'. What in LCS corresponds to ACTOR in (18) is not stated in (20) but at an extra tier of representation displaying force relations, a bit like the function of *effort* in Figure 1. The formats thus seem to allow for comparison in a possibly tractable manner, and possibly with the conclusion that the information which can be displayed in them is essentially the same (both can in principle also be enriched to display more fine-grained information).

While the information of SIT thus in principle unfold the universe of what can be represented within approaches such as Lexical Semantics or Conceptual Semantics, the attributes defined within the attribute ACTNT are extremely few, and reflect only an enumeration of the arguments which are grammatically manifest in the sentence in question, and their numbering (as ACT1, ACT2, etc., up to maybe ACT4) only reflects an ordering between them particular to that sentence or predicate, based vaguely on a dimension of agentivity or 'proto-role', and respecting a conception of stages in a derivation. Thus, although the sentence in (3) has three syntactic arguments, the ACT-roles are just ACT1-ACT2 relative to the *cause*-predicate, and ACT1-ACT2 relative to the *cook*-predicate, as displayed in (7), in both cases such that what would be a subject in a closest paraphrase is ACT1. In the case of a passive construction, the role ACT1 will belong to the 'highest-ranking' item in the conceivable active form, which is again to say the subject in this active structure, so that the subject in the passive version may represent ACT2 or ACT3 as a role. This is a design familiar in formal grammars, corresponding to what is called 'Semantic Argument Structure' in Grimshaw (1992) and related work, to argument structure as common in Predicate Logic, and with a semi-shallow robustness which makes it suitable for 'Deep' computational grammars such as those based on HPSG, where the level of logical representation called 'Minimal Recursion Semantics' (MRS; cf. Copestake et al. 2005) displays sentential semantic content in this form. The CL code does not directly display ACTNT structure, but given the GF specifications of all items, and the formal tractability of derivational structure as illustrated in (4)-(7)-(11) for the Bantu derivational form, the ACTNT roles of any derived sentence structure are tractable, and in plain non-derived structures the subject is the ACT1, the direct object the ACT2, and an indirect object the ACT3. For oblique objects one can use ACT4, ACT5, etc., or ACTobl.

Although superficially similar to the system using ARG1, ARG2, etc in PropBank,[6] the ACTNT system differs from that of PropBank in that none of the attributes ACT1, ACT2, etc represents a specific semantic value – ARG1 in PropBank, in contrast, is agent. In this way, semantic richness is represented in the SIT system, whereas the kind of semantics that very closely follows grammatical structure is represented in the ACTNT system.

Having now introduced the annotation system Construction Labeling, the components of the TFS to which it can be mapped, and in particular the semantic components, what we call the *grammatical interoperability* of the CL notation has been demonstrated. We now turn to uses and applications of the system.

5. Semantic annotation in valence lexicons and valence corpora

The CL annotation code is used in three types of applications – corpora, lexicons, and computational grammars, and in addition in the compilation of *language valence profiles*, which in a compact format represent the construction types and valence types available in a language.[7] We now describe the role of the semantic labels relative to such systems, first a construction and valence inventory of Ga, and then of Norwegian.

5.1 Situation types in a Ga construction and valence inventory

A valence resource for Ga was developed by Prof. Mary Esther Kropp Dakubu as an extension of the Toolbox lexicon underlying her Ga-English Dictionary Dakubu (2009). In this extension, valence specification using the CL code was added systematically, resulting in about 2000 entries such that each entry of a verb represents one valence frame. Each such entry is illustrated by a fully annotated sentence, which means that the lexicon is at the same time a valence corpus of about 2000 short sentences. An edited version of this resource is found at Ga Valence Profile in the downloadable text file Ga verb dictionary for digital processing', cited as Dakubu (unpublished a), to which we refer in the following; a larger extension is available in Dakubu (unpublished b).

Ga makes little use of prepositions and adjectives, so that constructions involving nouns and verbs may be seen as playing a relatively large role, the latter for instance

[6] Cf. https://propbank.github.io/; Palmer et al. (2005).

[7] See Ga Valence Profile, and with examples, on Ga Appendix. For a valence profile for Norwegian, see Verbconstructions Norwegian - all types. Further examples are Valence Profile Kistaninya, Valence Profile English, and Gurene verb constructions. An inventory of CL tags in total is found at https://typecraft.org/tc2wiki/Construction_Label_tags.

through *multiverb expressions* subsuming *Serial Verb Constructions (SVCs)*, *Extended Verb Complexes (EVCs*) which are sequences of preverbs preceding a main verb, heading a clause by itself or partaking in an SVC), and *Verbid Constructions (ViD*), where verb phrases play a role of adverbials.[8] The use of complex pre-nominal specifiers within noun phrases is another predominant feature, briefly summarized in terms of number of entries exhibiting them in Table 1 below, and exemplified in (12).

Table 1 Nominal specifications in terms of number of verb entry specifications

Bodypart relation	158
Identity relation	110
Subject headed by relational noun	99
Object headed by relational noun	690
Object's specifier headed by relational noun	29

The array of Situation Types used was conceived in parallel with the process of annotation of the data,[9] rather than being built on any pre-existing inventory. The frame types used in FrameNet[10] were consulted but found to be too English-biased for the purpose. As a result the situation types are at a somewhat general level, but also not very abstract – relative to types like those in Figure 1 they occupy the lower half, but not as far down as matching lexically-specific meanings; they thus may be said to classify *construction type meanings* rather than verb meanings. Table 2 renders the most frequently used type labels, ordered alphabetically and with indication of the number of entries exhibiting them:

Table 2 Situation Type labels most frequently applied.

ABSENT	29	MOTIONDIRECTED	55
ACQUISITION	29	PHENOMENON	29
CARETAKING	12	PLACEMENT	53
CAUSATIVE	23	POSTURE	7
CAUSED	17	PROPERTY	164
CLOSING	4	DYNAMIC-PROPERTY	13
COGNITION	83	PSYCHSTATE	23
COMMUNICATION	178	REMOVAL	47
COMPARISON	29	SENSATION	16
COMPLETED-MONODEVMNT	6	TRANSFER	47
CONTACT	56	USINGVEHICLE	5
CREATION	14		
CUTTING	19		
EJECTION	15		
EMOTION	29		
EXPERIENCING	45		
MAINTAINPOSITION	25		
MOTION	180		

With the set of 2000 entries classified by CL strings, one can investigate the frequency of frames used, the correspondence between syntactic and semantic structure, the clustering of certain valence types for sets of verbs (constituting 'Verb Classes', see below), and more. To exemplify, the layout of information illustrated in (21) indicates the entry ID, the full CL construction specification, and the gloss of the verb heading the construction. The ID links to a parallel display where the instantiating sentence is given. The entries exemplified all have the situation type MOTIONDIRECTED:

(21)

fa_212 := v-tr-obNomvL-suAg_obLoc-INCHOATION-MOTIONDIRECTED (appx. gloss: "start up")

fo_338 := v-tr-obPostp-suAg_obPath-REFLEXIVE-MOTIONDIRECTED (appx. gloss: "turn around")

ke_737 := sv_suAspID-suAg-v2tr-v2pv1Pro-v2obLoc-MOTIONDIRECTED (appx. gloss: "proceed")

kɔ_757 := v-intr-suAg-MOTIONDIRECTED (appx. gloss: "climb")

kpeleke_841 := v-tr-obPostp_obSpecThAbst-suAg_obTh-MOTIONDIRECTED (appx. gloss: "land")

The glosses of the 55 entries with this situation type involve the following items as gloss of the head verb, listed by number of occurrence:

(22)

"go" – 12, "come" – 7, "push away" – 6, "arrive at" – 3, "land" – 3, "go before" – 3, "start" – 2, "run" – 2, "visit" – 2, "forget/leave" – 2, "push" – 2, "climb" – 2, "repent/turn away from" – 2, "travel" – 1, "turn around" – 1, "proceed" – 1, "depart" – 1, "strike" – 1, "paddle" – 1, "trail" - 1

For one thing, this illustrates that Situation Types and lexical meanings are distinct. For investigations into verb classes, such numbers, paired with the grammatical structures of the constructions involved, provide a good starting point. As for the grammatical structures involved in the 55 entries, most total strings are unique, only one applies to 5 entries, one to 3 entries, and three to 2 entries. However, the CL code allows one to compare also with regard to substrings, which allows for a flexible methodology of establishing correspondences in these domains.

The literature on Valency Classes (aka Verb Classes) starts with Levin 1993, which is an attempt to find correlations between verb meanings and the arrays of valency frames available for given verbs.[11] Levin's approach has been pursued for English during *VerbNet* at a large scale,[12] which is a resource featuring more than 6000 verbs divided into nearly 300 verb classes. In the *The Leipzig*

[8] See Dakubu 2004a,b, 2008, Dakubu et al. 2007.

[9] Conducted by Prof. Dakubu, with a few consultants.

[10] https://framenet.icsi.berkeley.edu/fndrupal/ 'Frame' in the FrameNet system corresponds to what we here call situation type.

[11] For instance, for the 'spray-load'-alternation verbs in English, as exemplified in *spray paint on the wall* vs *spray the wall with paint*, a characterizing feature is the expression of two incremental dimensions at the same time (here the amount of paint and the area of wall covered), whereby either one or the other can be expressed by an NP inducing completeness of that dimension, reflected in the alternating frames (the 'non-completed' dimension represented by the PP).

[12] http://verbs.colorado.edu/~mpalmer/projects/verbnet.html).

Valency Classes project (Malchukov and Comrie 2015) and the accompanying valency database ValPaL[13] the arrays of frames for 80 verb meanings are compared across 30 languages; here one uses English verbs as 'names' of verb meanings – for instance, the 'kill verb' is treated as a constant entity 'KILL' across the languages. This gives, for each of the 80 verbs, a view of the frames that the verb can take across these languages.

The current enterprise is mostly in the spirit of VerbNet, since we are dealing with a large number of verbs. Apart from size differences, what is particular to the present approach is the way in which it allows the annotation code to serve as a key instrument of representation. In the VerbNet verb entry illustrated in Figure 2 below, the syntactic specification consists of a dependency tree (not shown here) and a line combining POS and semantic role in an order matching the linear order in which the relevant constituents occur. In our approach, a syntactic format linked to the linear order of the constituents in the analyzed string is given in the example sentences, while the ordering within the CL string is independent of linear order in the examples. The CL syntactic code nevertheless comes close to representing a dependency analysis (and a full-fledged syntactic and semantic parse can in principle be called upon[14]). The display under SEMANTICS has a richness of content comparable to our Situation Structure, but closer in style to the predicate-argument structure exemplified in (20) than to the AVM format used here.

« Jessica loaded boxes into the wagon. »

SYNTAX: Agent VERB Theme { PREP } Destination

SEMANTICS:
HAS_LOCATION(e1 , Theme , ?Initial_Location)
DO(e2 , Agent)
MOTION(ee3 , Theme , Trajectory)
¬ HAS_LOCATION(ee3 , Theme , ?Initial_Location)
CAUSE(ee3 , e2)
HAS_LOCATION(e4 , Theme , Destination)

FORCE DYNAMICS: Volitional Apply FD representation

Figure 2 Copy from VerbNet view of 'spray-9.7' (March 27, 2020)

The design used in VerbNet has counterparts in most other valence-related applications,[15] so on a comparative note, we may say that the present analytic apparatus offers counterparts to all of the representations found in the standard applications. What the CL notation provides in addition is a compact one-line view of all of the relevant factors brought together, and an algorithm by which this compact notation is linked to the analytically full representations.

A further comparative aspect lies in the use of hierarchical organization: FrameNet to a mild extent uses this for frames, i.e., situation structures, but not with the efficiency of TFS as illustrated in Figure 1. In return, though, a full system comprising all of the situation type labels in Table 2 (or the totality of the 140 types used) has not yet been constructed. VerbNet uses organization of entries like that for *spray-9.7* in Figure 2 such that a common meaning 'dominates' those structures that share that meaning. Such an organization can readily be provided also in the present notation. For instance, the array of entries for *ba* 'come' includes the following,

<ba_1, v-intr-suAg-MOTIONDIRECTED>
<ba_2, evSuAg-vintr-pv1obTh-MOTIONDIRECTED>
<ba_3, v-tr-obPostp-suAg_obLoc-MOTIONDIRECTED>
<ba_6, v-tr-suAg_obEndpt-MOTIONDIRECTED>
<ba_8, v-ditr-suAg_obTh_ob2Endpt-MOTIONDIRECTED>

which could be displayed as follows, keeping in mind that all labels unify, and hence a hierarchy (or ontology) can in principle be designed with any label as top node:

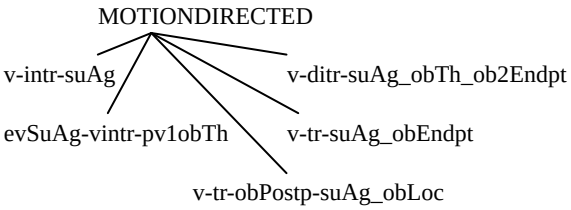


Figure 3 Hierarchical organization of entries for *ba* 'come' with *MOTIONDIRECTED* as common meaning

5.2 CL code in a verb valence lexicon and a valence corpus for Norwegian

A cluster of resources for Norwegian has been created where the CL code plays a pivotal role in (i) encoding the lexical types of verbs as represented in the HPSG-type computational grammar *Norsource*[16], (ii) constituting the valence specifications in a valence lexicon of about 13,000 valence entries, organized using the tags as lexical types coextensive with the verbal part of the grammar lexicon,[17] and (iii) serving as valence tags in a verb valence corpus generated by the grammar.[18] In the latter, valence and construction tags are thus assigned to verbs in 22,000 sentences, illustrated in Figure 4 below: here valence annotation labels supplement a standard IGT annotation,[19] with the CL notation accompanied by two other formats of valence labelling, 'SAS' for 'syntactic argument structure'

[13] http://valpal.info/

[14] Through a parser, cf. Hellan (2020).

[15] Among existing valence dictionaries are for instance: English:FrameNet; VerbNet; PropBank; German: Evalbu; Chech: Vallex; Polish: Walenty; respective urls: https://framenet.icsi.berkeley.edu/fndrupal, http://verbs.colorado.edu/~mpalmer/projects/verbnet.html, https://ufal.mff.cuni.cz/czengvallex, http://hypermedia2.ids-mannheim.de/evalbu/, http://ucnk.ff.cuni.cz, http://clip.ipipan.waw.pl/Walenty.

[16] *NorSource* (cf. Hellan and Bruland 2015) has been maintained since 2001. Code files are downloadable from GitHub: https://github.com/Regdili-NTNU/NorSource/tree/master.

[17] The valence lexicon, called *NorVal*, is under development, cf. Quasthoff et al. (2020). An earlier version can be seen at http://regdili.hf.ntnu.no:8081/multilanguage_valence_demo/multivalence, called *MultiVal*, described in Hellan et al. (2014), where four lexicons based on computational HPSG grammars for Norwegian, Ga, Spanish, Bulgarian are brought together using the same types of valence frame labels.

[18] Cf. Hellan et al. (2017, forthcoming).

[19] Using the glossing system and interface of TypeCraft (https://typecraft.org/; cf. Beermann and Mihaylov (2014)).

and ‘FCT’ for ‘functional label’; the valence frames of both the main verb (*vite* (‘know’)) and the subordinate verb (*forbause* (‘surprise’)) are specified:

String: Jeg vet at hun forbauset Ola
Free translation: I know that she surprised Ola

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

vite: SAS: NP+Sdecl
FCT: transWithSentCompl
CL: v-tr-obDECL

forbause: SAS: NP+NP
FCT: transitive
CL: v-tr

Figure 4 Sample representation of valence annotation in a valence corpus combined with morphological glossing

To the CL expressions, which here lack role and Situation Type labels, one could easily add such labels, but in the current state their absence reflects the cirumstance that most of the valence specifications in the underlying grammar, which uses CL code, do not have such specifications. Among reasons why this is so is that the introduction of semantic features in a large scale grammar is not only time consuming in linguistic respects but also requires the balancing of combinatorial complexities arising with the introduction of new dimensions of specification. Within limited domains like that of *location* and *direction* it has however been done, through the specification of lexical items and combinatory rules of the relevant kinds. Here a large part of the specifications are tied to prepositions and adverbs, so that for instance in the lexical specifications operative for the sentence *De løper til byen* (‘they run to the town’), *løpe* is encoded as intransitive directional and *til* is encoded as an end-point preposition. The resulting parse will in Situation Type terms have the label MOTIONtoENDPOINT, while the verb by itself belongs to the type MOTIONDIRECTED. To the extent that a display like that in Figure 4 is generated through a grammar, thus, it cannot represent the verb as such as being of the type MOTIONtoENDPOINT. This illustrates a further factor by which a grammar-generated corpus can fail to be as specific as a ‘hand-made’ corpus. Still, the CL formalism being one where semantic specifications *can* be seamlessly added to the grammatical ones, it leaves room for incrementally adding such specifications in the corpus.

Figure 5 illustrates this interplay between verb and preposition. In the NorSource parse tree to the left one sees the critical specifications of the items *løpe* and *til*, and in the MRS to the right the specification ‘Role: endpoint’ is induced through the semantic specifications of the preposition and the verb together:

head-subject-rule
de_perspron
de
telic-pp-mod-vp-rule
pres-infl_rule
løpe_***intrdir***_vlxm
løper
head-prep-comp-rule
til_***dirtel-end***-**p**
til
sg_def_m_final-full_irule
sg-masc-def-noun-lxm-lrule
by_mascanim_nlxm
byen

ltop=h0, index=e1
h3:de_pron_rel([arg0:x2])
h4:_pronoun_q_rel([arg0:x2, rstr:h5, body:h6])
h7:***_løpe_v-intr_rel([arg0:e1, arg1:x2***])
h7:***_til_p-dirtel_rel([arg0:u8, arg1:x2, arg2:x9,*** iarg:u10])
h11:_by_n_rel([arg0:x9])
h12:_def_q_rel([arg0:x9, rstr:h13, body:h14])
< qeq(h5,h3), qeq(h13,h11) >
e1, sort=verb-act-specification, sf=prop, e.tense=pres, e.mood=indicative, e.aspect=semsort x2, wh=-, png.ng.num=plur, png.pers=thirdpers, role=***mileage-obj***
u8, sort=verb-act-specification
x9, wh=-, bounded=+, png.ng.num=sing, png.ng.gen=m, png.pers=thirdpers, role=***endpnt***

Figure 5 NorSource parse tree and MRS for *De løper til byen* (‘they run to the town’), with encodings relevant for directionality in italic boldface. (Copied from the web demo http://regdili.hf.ntnu.no:8081/linguisticAce/parse on March 20, 2020.)

The MRS construction is based on what corresponds to the ACTNT component described in section 4 (but here with ‘ARG’ rather than ‘ACT’), thus a fairly shallow level of semantic description, however with the possibility of specifying the ARG/ACT for semantic roles, which is done in the grammar, although at an earlier point.[20]

6. Labeling for scope

Here we consider a possible extension of the CL style of specification to phenomena standardly analyzed in terms of *scope*. First addressing quantifier scope,[21] we may build on the CL designs used for NP internal structures, illustrated in section 3. In a sentence like (23), one commonly recognizes two scoping possibilities, for which the CL-style strings in (24) provide a labeling, with *QS* understood as ‘quantifier out-scoping’; (a) represents *two men* as having wide scope, with *suQSob* read as ‘subject outscoping object’, and (b) represents *every book* as having wide scope:

(23) Two men read every book.

(24) a. v-tr-suQSob
b. v-tr-obQSsu

In the more complex (25), plausible scope relations are probably restricted to those in (26), with *adj* interpreted as ‘adjunct’, here *every evening* (thus, any construal implying a man as reading a book over again counts as implausible):

(25) Two men read every book every evening

(26) a. v-tr- suQSob_adjQSsu
b. v-tr-obQSsu_adjCSob

[20] Cf. Beermann and Hellan 2004, Hellan and Beermann 2005.

[21] See Bunt (2020) for an overview of issues relating to the annotation of quantifier scope.

A notation like this may be useful for corpus annotation with the goal of finding patterns as to when multiple scopings are possible. Given that syntactic subjects can probably by default be counted as outscoping everything that they c-command, the (a) versions of (24) and (26) may count as redundant. The link into the AVM formalism can follow the design of the feature structure input to MRS representations in HPSG grammars, as outlined in Copestake et al. 2005.

Although quantifier scoping is per se perhaps a strictly semantic matter, the participants in scoping relations are generally syntactically identifiable,[22] which makes the general design of the present notation possible.

Among phenomena manifest in a wider domain of configurations is 'reported speech', as studied (a) in their role in determining morpho-syntactic patterns across languages, for instance in phenomena like subjunctive mood and logophoricity,[23] and (b) in their role in various kinds of apparent analytic paradoxes in formal representation. A common denominator of many instances of both types is the choice of *whose construal* is reflected in the piece of text concerned: either the construal of the wording as that of the *speaker*, or the construal of the wording as that of one of the *participants*, mostly the *subject*. A typical example from the (b) domain is (27),

(27) John thinks that the statue is taller than it is.

where the wording *taller than it is* is most reasonably attributed to the speaker, not to John. Exploring the annotation format used above, *speaker construal* relative to a text piece '…' can conceivably be annotated as 'spkCS…', with *spk* for 'speaker', and subject construal as 'suCS…', with *su* as before, and *CS* in both cases understood as *construal scope*. The fruitfulness of the format may depend in part on how easily what is indicated by '…' can be identified for given constructions. As exemplified in (13) above, the notation allows for the specification of paths 'down' into constituents, and in principle, as long as a text piece coincides with what can be syntactically motivated as a constituent, a path can be defined; (28) illustrates the point for *speaker construal* relative to the example in (27), for a path specification into the object clause's predicate (marked as *sc*, for 'secondary predicate').

(28) spkCSobDECLsc

(This reads as: 'speaker has construal-scope over the secondary predicate of the declarative clause constituting the (matrix) object (counting "taller than it is" as secondary predicate).)

It will be a natural task to explore such extensions of the code, also transcending the sentence as annotation domain.[24]

7. Conclusion

The semantic annotation system presented is an integral part of a grammatically complete annotation system, used both in corpus annotation, verb valence lexicons and formal and computational grammars. It is linked to a Type Feature Structure system sustaining formal grammars in general, and in the present system with a component of *Situation Structure* as an integral part. This component content-wise represents what is often referred to as lexical or conceptual semantics, but unlike most formal systems in this domain, the present version is constructed fully in terms of Typed Feature structures, whereby it has been fully integrated with the overall grammatical system. Apart from the formal interest in constructing such an architecture, the integration also gives formal expression to the circumstance that meaning, as the subject of semantics as a linguistic field, is inextricably carried by grammar, the co-construal of semantics with grammar thus being a desideratum of any formal framework of language. Thus, although representations within Situation Structure can be viewed by themselves, aspects which have a grammatical exponence can be represented with an explicit link to the exponence factor (where grammatical functions are main 'navigation points' relative to grammatical structure).

That being said, outlining the algorithmics of such a co-construal in principle is one thing, realizing it in a large scale representation of a language is another; our description of the resources for Ga suggest that this is fully possible. The Construction Labeling (CL) formalism for annotation can help in attaining significant coverage of linguistic material, as it can be used on a purely descriptive basis (thus not in tandem with formal analysis), and especially when done in parallel with (or posterior to) more elementary grammatical analysis and glossing. This is what has been demonstrated for Ga. For Norwegian we have demonstrated that the CL code can be used in an effective interplay between grammar, valence lexicon and valence corpus, providing language-wise full scale analytic structures to which situation structure semantic information can be incrementally added.

What has here been outlined resides partly in work done over the last decade, but with the formal integration of the CL system with the grammatical type representation as a novel step. With sentence analysis and sentence annotation being consolidated, we have indicated directions in which the annotation formalism can be brought into scopal analysis, and hopefully next into the analysis of larger text units representing further dimensions of analysis.

The compactness of the code facilitating the annotation of large corpora for valence- and construction type, and for the construction of large valence lexicons, this holds not only from the perspective of attaining complete coverage relative to a given language, but also from cross-linguistic perspectives concerning the presence of given construction-/valence types across languages. These are the main perspectives for the use of the annotation system into cross-linguistic construction-and-valence description and typology.

[22] Wide scope readings of implicit arguments and null pronouns are not commonly encountered.

[23] Cf. Nikitina (2019).

[24] A medium of text representation where examples can be searched relative to *strings* of annotation code (as, e.g., in TypeCraft valence specifications, as exemplified in Figure 4), could allow for a search query such as 'suCSobDECL', which would lead to all examples annotated for subject construal into a declarative clause, including those annotated with 'spkCSobDECLsc' and for other embedded constituents as well.

8. Bibliographical References

Beermann, Dorothee and Lars Hellan. 2004. A treatment of directionals in two implemented HPSG grammars. In Stefan Müller (ed) *Proceedings of the HPSG04 Conference*, Katholieke Universiteit Leuven. CSLI Publications /http://csli-publications.stanford.edu/

Beermann, Dorothee and Mihaylov, Pavel. 2014. Collaborative databasing and Resource sharing for Linguists. *Languages Resources and Evaluation* 48. Dordrecht: Springer, 1-23.

Bresnan, Joan. 2001. *Lexical Functional Grammar*. Oxford: Blackwell.

Bunt, Harry. 2020. Semantic Annotation of Quantification in Natural Language. Tilburg Centre for Creative Computing.

Copestake, Ann. 2002. *Implementing Typed Feature Structure Grammars*. CSLI Publications.

Copestake, Ann, Dan Flickinger, Ivan Sag and Carl Pollard. 2005. Minimal Recursion Semantics: An Introduction. *Journal of Research on Language and Computation*. 281-332.

Dakubu, M.E. Kropp, 2004a. The Ga preverb *kɛ* revisited. In Dakubu and Osam, eds., *Studies in the Languages of the Volta Basin* 2: 113-134. Legon: Linguistics Dept.

Dakubu, M.E. Kropp, 2004b. Ga clauses without syntactic subjects. *Journal of African Languages and Linguistics* 25.1: 1-40.

Dakubu, M.E. Kropp, 2008. Ga verb features. In Ameka and Dakubu eds., *Aspect and Modality in Kwa Languages.* Amsterdam & Philadelphia: John Benjamins Publishing Co. Pp. 91-134.

Dakubu, Mary Esther Kropp, 2009. *Ga-English Dictionary with English-Ga Index*. Accra: Black Mask Publishers.

Dakubu, Mary Esther Kropp. Unpublished a. 'Ga verb dictionary for digital processing', accessed at https://typecraft.org/tc2wiki/Ga_Valence_Profile

Dakubu, Mary Esther Kropp. Unpublished b. Ga Verbs and their constructions. Monograph ms, Univ. of Ghana.

Dakubu, M.E.K., L. Hellan and D. Beermann. 2007. Verb Sequencing Constraints in Ga: Serial Verb Constructions and the Extended Verb Complex. In St. Müller (ed) *Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*. Stanford: CSLI Publications. (/http://csli-publications.stanford.edu/)

Dakubu, Mary Esther Kropp, and Hellan, Lars 2016. Verb Classes and Valency Classes in Ga. Presented at SyWAL II (Symposium on West African Languages), Vienna.

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 Publ. Co.

Grimshaw, Jane. 1992. *Argument Structure*. Cambridge, Mass: MIT Press.

Hellan, Lars. 2019a. Construction-Based Compositional Grammar. March 2019. *Journal of Logic Language and Information.* DOI: 10.1007/s10849-019-09284-5

Hellan, Lars. 2019b. Situations in Grammar. In Essegbey, J., Kallulli, D. and Bodomo, A. (eds). *The grammar of verbs and their arguments: a cross-linguistic perspective*. Studies in African Linguistics. Berlin: R. Köppe.

Hellan, Lars. 2020. A computational grammar for Ga. LREC 2020, the RAIL workshop.

Hellan, Lars and Dorothee Beermann. 2005. Classification of Prepositional Senses for Deep Grammar Applications. In: Valia Kordoni and Aline Villavicencio (eds.). *Proceedings of the 2nd ACL-Sigsem Workshop on The Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications.*

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., D. Beermann, T. Bruland, M.E.K. Dakubu, and M. Marimon. 2014. *MultiVal*: Towards a multilingual valence lexicon**.** In Calzolari, Nicoletta et al. (eds.) Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014), pages 2785–2792, Reykjavík, Iceland. ELRA.

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.

Hellan, L., Beermann, D., Bruland, T., Haugland, T., and Aamot, E. 2017. Creating a Norwegian valence corpus from a deep grammar. In: Vetulani et al. (eds) *Proceedings from The Language & Technology Conference* (LTC) 2017, Poznan.

Jackendoff, Ray. 1990. *Semantic Structures*. MIT Press.

Malchukov, Andrej L. & Comrie, Bernard (eds.) 2015. *Valency classes in the world's languages.* Berlin: De Gruyter Mouton. 2015.

Nikitina, Tatyana. 2019. The mysteries of reported speech. Workshop on *Reported discourse across languages and cultures*. LLACAN, CNRS; Paris.

Palmer, Martha, Dan Gildea, Paul Kingsbury, The Proposition Bank: A Corpus Annotated with Semantic Roles *Computational Linguistics Journal, 31:1*, 2005.

Pollard, Carl and Ivan Sag. 1994. *Head-driven Phrase Structure Grammar*. Chicago Univ. Press.

Quasthoff, Uwe, Lars Hellan, Erik Körner, Thomas Eckart, Dirk Goldhahn, Dorothee Beermann. 2020. Typical Sentences as a Resource for Valence. LREC 2020.

Smith, Carlota. 1991, 1997. *The parameter of aspect*. Dordrecht: Kluwer.

Vendler, Zeno. 1967. *Linguistics in Philosophy*. Ithaca, NY: Cornell Univ. Press.

Verkuyl, Henk. 1996. *A Theory of Aspectuality*. Cambridge University Press.

9. Language Resource References

CL code: Construction Label tags

Norwegian valence corpus:
https://typecraft.org/tc2wiki/Norwegian_Valency_Corpus

Multilingual valence resource :
http://regdili.hf.ntnu.no:8081/multilanguage_valence_demo/multivalence

Ga valence profile and lexicon files:
https://typecraft.org/tc2wiki/Ga_Valence_Profile