MultiVal – towards a multilingual valence lexicon

Lars Hellan[a], Dorothee Beermann[a], Tore Bruland[a], Mary Esther Kropp Dakubu[b], Montserrat Marimon[c]

[a]NTNU, N-7491 Trondheim, Norway
[b]University of Ghana, Legon, Accra, Ghana
[c]Universitat Pompeu Fabra, Roc Boronat 138, 08018-Barcelona, Spain
Email: {lars.hellan, dorothee.beermann, torebrul}@ntnu.no, medakubu@gmail.com, montserrat.marimon@upf.edu

Abstract

MultiVal is a valence lexicon derived from lexicons of computational HPSG grammars for Norwegian, Spanish and Ga (ISO 639-3, gaa), with altogether about 22,000 verb entries and on average more than 200 valence types defined for each language. These lexical resources are mapped onto a common set of discriminants with a common array of values, and stored in a relational database linked to a web demo and a wiki presentation. Search discriminants are 'syntactic argument structure' (SAS), functional specification, situation type and aspect, for any subset of languages, as well as the verb type systems of the grammars. Search results are lexical entries satisfying the discriminants entered, exposing the specifications from the respective provenance grammars. The Ga grammar lexicon has in turn been converted from a Ga Toolbox lexicon. Aside from the creation of such a multilingual valence resource through converging or converting existing resources, the paper also addresses a tool for the creation of such a resource as part of corpus annotation for less resourced languages.

Keywords: multilingual valence lexicon, HPSG grammars, less-resourced language tools

1. Introduction

The present paper presents a partial valence repository, populated with lexical information from three languages, namely Norwegian, Spanish and Ga (ISO 639-3, gaa). Our ultimate goal is the construction of a repository where one can identify cross-linguistic *valence- and situation type pairs – VSP*s. For instance, there is a match between English *put* and the Ga verb *wo*, exemplified in (1) below:

(1)	**Amɛ-wo**	**tsɔne**	**lɛ**	**mli**	**yɛlɛ**
	3P.AOR-put	vehicle	DEF	inside	yam
	V	N	Art	N	N

Close transl: 'They put [vehicle's inside] [yam]'
Free transl.: 'They put yams in the lorry.'

By means of a VSP inventory we want to be able to identify *PLACEMENT* as a situation type characterizing both *put* and *wo*, and from a look-up in the Ga VSP inventory, we are able to suggest a valence frame of the type instantiated in (1).

The benefits of such a repository for applications in translation, language teaching, and other, are obvious; our focus in this paper is to describe the construction of a repository that takes some important steps towards such a goal, called *MultiVal*. [1] This repository presently contains fairly detailed, unified information about formal and functional aspects of the valence patterns of the three languages. Situation type is so far more rudimentarily represented.

The project relates in different ways to contemporary initiatives such as FrameNet, VerbNet, ValPaL, and ImagAct.[2] Among previous initiatives with very similar goals may be mentioned the European project PAROLE (LE-4017), which was the first project producing corpora and lexicons in many languages (Catalan, Danish, Dutch, English, French, German, Greek, Italian, Portuguese, Spanish, and Swedish), built according to the same design principles, linguistic specifications, and representation format.[3] Although not directly continuing any of these initiatives, the application MultiVal is an exercise in deriving new uses from already existing resources, as will be described below. .

2. The functionalities of *MultiVal*

2.1 Representing *Argument Frame*

The repository is primarily organized according to *argument frames*. By an *argument frame*, we understand a pattern of sentence constituents which typically appears surrounding the main verb of a sentence.[4] The notion in principle includes the following:

[1] The web demo and its wiki portal are, resp.: http://regdili.idi.ntnu.no:8080/multilanguage_valence_demo/multivalence , http://typecraft.org/tc2wiki/Multilingual_Verb_Valence_Lexicon. Different stages and aspects of the project are reported in Hellan and Bruland 2013 and Hellan et al. 2013.

[2] http://verbs.colorado.edu/~mpalmer/projects/verbnet.html, http://www.eva.mpg.de/lingua/valency/, www.imagact.it

[3] The model was based on the EAGLES recommendations for morphosyntactic information and verb syntax and on the extended GENELEX model.

[4] The circumstance here to be communicated is sometimes phrased as 'be necessitated by the verb', but a verb may well have more than one argument frame, and rather than saying 'disjunctively necessary' or the like, we use the notion 'typical'.

(a) 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;
(b) semantic argument structure, that is, how many participants are present in the situation depicted, and which roles they play (such as 'agent', 'patient', etc.);
(c) linkage between syntactic and semantic argument structure, i.e., which grammatical functions express which roles, and possible roles not expressed; here also belong identity relations, part-whole relations, etc., between arguments;
(d) aspect and Aktionsart, that is, properties of a situation expressed by a sentence with the valence in question, in terms of whether it is dynamic/stative, continuous/instantaneous, completed/ongoing, etc.;
(e) type of the situation expressed, in terms of some classificatory system.

Thus, an argument frame can in principle be represented as an n-tuple of these factors. The present repository has *discriminants* constituting a subset of these factors, which we now describe.

The only factor which is reasonably well consolidated across frameworks is (a), and so this is the cornerstone of the present system. Even this factor can be described in many ways, and frameworks differ as to how it should be done. Accommodating the latter fact, we employ two formats for identifying a syntactic argument frame, one – called *syntactic argument structure (SAS)* - residing in a sequence of constituent labels in a notation perhaps most grounded in generative grammar (e.g., "NP+NP" for two NPs as constituents), and one – called *functional label (FCT)* - through a single label providing an over-all characterization of the frame, rooted in widespread grammatical tradition (e.g., 'transitive'). These are the main *discriminants* of the present system, thus representing two angles at the representation of factor (a).

Although oriented according to so-called 'formal' properties of the constituents of the frame, that is, head projections, the SAS notation does mark whether an NP is an argument-bearer or a predicative; it also indicates for infinitives whether they are controlled or not; and in some salient cases of selected prepositions or similar, the word itself is mentioned. The specification is essentially neutral relative to the SVO-parameter, in that the position of 'V' is not mentioned, hence "NP+NP" could be used for whatever position the verb has relative to the NPs. Around 158 patterns are so far defined at this level for Norwegian, 120 for Spanish, 40 for Ga. FCT notions (like 'transitive') are also word-order neutral. Around 88 notions are so far defined at this level for Norwegian, 130 for Spanish, 20 for Ga.

We expect that any argument frame can be classified according to the SAS and FCT terms, whereas for the other types of factors, the discriminants of the system must leave room for the possibility that no appropriate value is found for a given frame. These discriminants carry the names 'SIT' for situation type, and 'Aspect' for aspect and Aktionsart. 'SIT' includes information about '-arity' of the logical relation expressed, and could in principle also have role information, but this is not done for the present. [5] Among the factors (a)-(e), the one not reflected is (c), i.e., 'linking' between syntactic and semantic level; SIT encompasses both (b) and (e).

Conceptually, we may view the discriminants of an argument frame as part of a matrix like in Table 1:

Carrying verb	**SAS**	**FCT**	**SIT**	**Aspect**	Instantiation
Verb	***X***	***Y***	***Z***	***W***	*sentence*

Table 1: Schematic view of the argument frame matrix

What *instantiates* an argument frame is a *sentence*, whereas the *carrier* of a given frame is a *verb*,[6] and these must in general be available for illustration of an argument frame type, represented by the non-red parts of the table. These discriminants are in principle all independent of the others: a given verb may occur in many argument frames, a given SAS can be paired with more than one semantic specification, and often also with more than one functional label, and so on. This is reflected in such a matrix: any combination of values can be specified.

In a multilingual repository it is crucial that all values under each category are defined language independently; this secures comparability between the data from the three languages. Reflecting properties of the grammars involved, it will still be such that each language employs only a subset of the total set of values defined for each category. An overview of the SAS and FCT values available for each language is found on URL…

2.2 The *MultiVal* search interface

The circumstance that each language employs only a subset of the total set of values defined for each discriminant, is reflected in the set of options which is offered on the search menu. For a query we can supply values under any combination of discriminants, and the prompt for a set of discriminants gives us specifications of those *carrying verbs* that fit the combination, for the language(s) specified. In addition to the discriminants SAS etc., the search can also specify a carrying verb, or a substring of a verb name starting from the left.

The *result* of a query consists of one or more specifications of *carrying verbs*, relative to the properties of argument frames indicated. These specifications are always complete, reflecting all the properties associated with the verb in the database. Figure 1 is an example of a search result showing the relevant properties of a verb in

[5] Also ontological information belongs here. For instance, when a prompt for Norwegian *gå* yields the English *go* and *walk*, based on the shared VSP 'intransitive with adverbial expressing directional motion', the correct selection is likely to be connected to properties of the subject; in this case, the Norwegian verb *gå* can have both vehicles and humans as subject, while the choice between *walk* and *go* is sensitive to animacy (and the use of legs).

[6] In principle, a given frame will normally have many possible carriers, in the form of many verb lexemes (and of course infinitely many instantiations).

Ga:

Lexicon Instance

Language	ga
Verb Id	bɔ_74
Syntactic Arguments	NP+NP+NP
FCT	ditransitive
SIT	ternaryRel
Aspect	
Verb Type	v-ditr
Example of type	
Orthography	<"bɔ">
Phon	<"bɔ̀">
Engl-gloss	<"create">
Example	E-bɔ mi wɔŋ
Gloss	3S.AOR-do 1S god
Free-transl	she invoked a deity against me.

Figure 1. View of the verb *bɔ* in Ga as a search result

What here corresponds to *Instantiation* in Table 1 is the set of lines Instantiation, IGT-GlossEngl, and FreeTranslEngl, and the *carrying verb* is specified by the lines Verb Id, EnglGloss, Orthography, whereas the *discriminants* realized by this result are stated in the lines initiated by 'SAS', 'FCT' and 'SIT'. (It is possible to also search for just a verb, and get a result also on the form in Figure 1.)

The actual search interface is not very different from the conceptual view in Table 1, however, since 'Instantiations' are hardly entities relative to which one can define a search (being full sentences), there is no field for Instantiation in the search interface. On the other hand, the search interface includes the discriminant *Type*, cf. Table 2 below. This is a specification of an argument frame relative to the 'provenance' resource, to be explained in the next section; here we also explain the notion 'Verb Id' seen in Figure 1. In Table 2, all of ***X, Y, Z, W, T*** are drop-down menus with options specific to the discriminant in question, and declared for the language in question:

Carrying Verb	**SAS**	**FCT**	**SIT**	**Aspect**	**Type**
Verb	***X***	***Y***	***Z***	***W***	***T***

Table 2: Schematic view of the query interface

In principle, it could be possible to design such a system to allow queries for, e.g., the set of FCTs that go with a given SAS in a language, or other results different from carrying verbs. At this stage, however, such possibilities have not been implemented.

3. The basic resources

As mentioned in the Introduction, this valence resource is constructed on the basis of other resources, namely *computation grammars* of the three languages, based on the framework *Head-driven Phrase Structure Grammar* (HPSG; Pollard and Sag (1994), Sag et al. (2003)). The grammars are, in the order in which they were entered into this database, the Norwegian grammar *NorSource*,[7] the Ga grammar *GaGram*,[8] and the Spanish grammar *Spanish Resource Grammar* (SRG).[9]

HPSG is a strongly 'lexicalistic' framework, i.e., one that encodes the argument structure properties of a construction in the description of the *head* of the construction, i.e., typically, the verb. Lexicons of broad coverage grammars in this framework thus contain much valence information for verbs, and constitute a potential resource also for a cross-linguistic valence representation, being designed in a common formalism, and with a uniform interpretation of the analytic notions employed. The present project may be seen as an effort in actually producing a cross-linguistically aligned resource from such 'pre-harmonized' language-particular resources.[10]

The branch of computational grammars in the HPSG framework relevant to the present project, was developed through the *LinGO* initiative at CSLI, Stanford, using the *LKB platform* (Copestake 2002), which is a general platform for typed feature-structure (TFS-) grammars. First of these grammars was the *English Resource Grammar (ERG)*, started in the 80ies, followed by a Japanese grammar and a German grammar; essential to the development of further grammars in the family was the 'HPSG Grammar Matrix' ('the Matrix'; see Bender et al. 2010), which was mainly based on ERG, and had its first phase of deployment during the EU-project DeepThought (2002-4). This design has integrated in it a format of semantic representation (independent in origin) called *Minimal Recursion Semantics* ('MRS'; cf. Copestake et al. 2005). The grammar family is currently supported by the DELPH-IN consortium (http://moin.delph-in.net/).

Common features of these grammars specially relevant to the present project are their *type* systems, in particular the sub-systems of *lexical types*, and the organization of *lexical entries*. As is general for TFS grammars, a type (apart from the 'top' type) is declared as a *subtype* of one or more other types, and may in addition have feature specifications more specific than the specifications of the corresponding features of the 'mother(s)'. The specification of a *lexical entry* is technically the definition of a subtype of a given *lexical type*, where specifications of the following features (attributes) are typically introduced: 'STEM' orthography, a semantic 'PRED' value, and – if the item is a lexeme – an indication of which inflectional paradigm it belongs to. For instance,

[7] http://typecraft.org/tc2wiki/Norwegian_HPSG_grammar_NorSource.
[8] For general information, see Hellan 2007.
[9] See Marimon (2013).
[10] To the extent that this enterprise is successful, this in turn attests to the role that theoretically based frameworks can play in digital resource building.

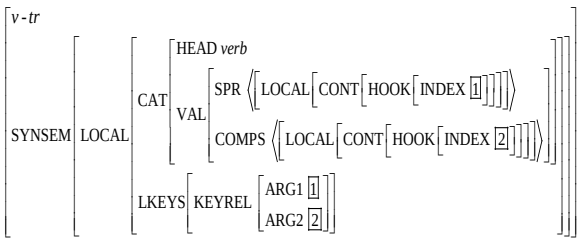
schematically using an English example, an entry for the transitive version of *kick* will have the form of a type definition of the type *kick_vtr*, declared as a subtype of the lexical type 'transitive verb', here rendered as *v-tr*. To illustrate, ':=' in (2) reads as 'is a subtype of', and the attribute specifications are those that distinguish this lexical entry from other lexical entries belonging to the same type 'transitive verb':

(2) kick_vtr := v-tr &
[STEM <"kick">,
INFLECTION paradigm_p,
PRED "_kick_v-tr_rel"].

The 'lexical entry' for the transitive verb *kick* is thus formally construed as a subtype of a more general lexical type *v-tr*; the label for this subtype – *kick_vtr* – is at the same time referred to as the *entry identifier*, and corresponds to what is called 'Verb Id' in Figure 1.

A type such as *v-tr* is associated with a feature structure, part of which in a standard HPSG design will look as in (3) below; it is in such structures that most of the information tied to lexical items of type 'transitive verb' sits, albeit this is information shared by all members of the type. The features SPR and COMPS introduce the syntactic items required in such a verb's valence, whereas KEYREL introduces the lexical semantic specification, with the attributes ARG1 and ARG2 representing an enumeration of the participants of a situation expressed by such a verb, and the indices indicating which participant is expressed by which syntactic item. Additional features in such a structure can express roles of the participants, aspect, and the situation type beyond the '-arity' information reflected in ARG1, 2, ..., all factors that in case would be reflected in a type label more complex than *v-tr*.

(3)



Moreover these types, and the feature structures they project, can classify verbs according to the syntactic category of their arguments, which include NPs, PPs, ADJPs, ADVPs, and CPs. Valence frames can be further constrained in terms of: optionality (of complements, as well as of marking preposition and of the complementizers introducing finite completive clauses), preposition classes for verbs of location and verbs of movement (constraints on the marking prepositions that are allowed to co-occur with verbs are set on the lexical items), control and raising relations, mood (indicative or subjunctive) of clausal subjects and complements, pronominal clitics, and, finally, those frame alternations that in the grammar are handled by means of lexical rules.

As is clear, the grammars encode information of all the types reflected in the discriminants mentioned above, and the population of the database is based on each language's set of verb types and set of verb entries. The exact procedure for this population is described in section 4. We first briefly comment on some particular properties of the three grammars' verb systems.

3.1 Verbs and verb types in SRG and NorSource

The Spanish Resource Grammar has about 8000 verb entries and 236 verb types, while Norsource has about 12500 entries and 348 verb types. These differences to some extent reflect a formal factor and not a difference in coverage, in that verbal types in the grammars differ as to whether they reflect *optionality* of arguments: SRG can subsume, through the marking of an item as 'optional', e.g., both a transitive and an intransitive frame of a verb in one and the same type and stated in one and the same entry, whereas NorSource consistently has one specific frame for each verb type (in both grammars opening for an item to undergo lexical rules).

The verb type coding itself is also different in the two grammars, with Norsource adopting the *Construction Label* system (Hellan and Dakubu (2009, 2010), Hellan (2008)). The respective full sets of these types are presented in the roll-down menu *Type* in the search interface of MultiVal.

The Norsource verb inventory is partly based on previously created resources for Norwegian, adapted for use in the grammar, [11] while the Spanish inventory has been built fully by the developer.

3.2 Verb types in GaGram

The Ga Grammar has about 2000 verb entries and, in the semantically non-enriched version, 144 verb types. It may be noted that Ga has at least three types of *multiverb constructions* (see, e.g., Dakubu et al. 2007), signalled in the *Type* drop-down menu by the substrings *sv-* (for 'serial verb'), *ev-* (for 'extended verb complex'), and *–Vid* (for 'Verbid').

The general layout of the Ga lexicon follows the Norsource layout in lacking optionality marking, and using the Construction Label system (cf. above). As opposed to both SRG and NorSource, however, the array of attributes in a Ga lexical entry is far larger than the format indicated in (2), reflecting its provenance from a Toolbox lexicon. In the following we comment on this provenance. [12]

The Toolbox program is provided with a large number of field names useful for many lexicographic purposes. It is also possible to add to this set. For the purpose of creating the Ga verb database, which was itself derived from a general dictionary of the Ga language created in

[11] The system TROLL (Hellan et al. 1989) and the successor NorKompLex.

[12] Converting Toolbox lexicons into LKB grammars was pioneered by Hirzel (2006), for an earlier version of GaGram; for another later instance, see Bender (2012).

Toolbox,[13] the system was enriched with a number of fields specifically designed to reflect the valency-related fields or "slots" of the Construction Label system (Hellan and Dakubu 2010). This means that in addition to the usual fields like \lx for lexeme, \ps for part-of-speech, \xv for example, etc., six additional fields were included. \sl1 corresponding to "slot 1" of the Construction Label system, contained the Head Specification (v in all relevant entries); \sl2 (slot 2 in CLS) provided valency type, including intr(ansitive), tr(ansitive), di(transitive); \sl3 (corresponding to slot 3 in Hellan and Dakubu 2010) gave the syntactic constituents and their properties – su(bject), ob(ject), obl(ique) and several others. \sl4 provided the thematic roles of each constituent (eg su(bject)Ag(ent), while \sl5 and \sl6 were provided for Aktionsart and Situation Type respectively. The last two were the least fully developed, and only the first three were in fact used in the conversion to the LKB grammar GaGram. Another innovation in the Toolbox system was the use of \xg as a field label for the interlinear glossing of the example sentence.

In going from the Ga Toolbox file to the lexicon file of the Ga grammar, we use the Toolbox fields: \lx, \ps, \ge, \ph, \sl1, \sl2, \xe, \xv, and \xg (more than one series of \xv, \xg and \xe can be present in the Toolbox file, only the last series is used). Each lexicon entry gets its own unique number. For instance, from the Toolbox entry in Table 3, through the correlation key indicated in the right-hand side of Table 4, we get the entry in the left-hand side of Table 4:

\lx ba	\pdl neg. fut	\xv E-ba oya
\hm 1	\pdv bang	\xg 3S.AOR-come quickly
\ph ba	\pdl imper	\xe he came quickly.
\ps verb annotated	\pdv bá	\xv È-bá-aa
\ pdl neg. imperf	\ge come	\xg 3S-come-NEG.IMPERF
\pdv baaa	\sl1 v-	\xe he didn't come; he didn't measure up
\pdl neg.perf	\sl2 intr-	\dt 27/Dec/2009
\pdv bako	\sl6 MOTIONDIR	

Table 3. Enumeration of fields in a Ga Toolbox entry

How the Toolbox entry in Table 3 comes out in LKB format	General schema of how Toolbox fields populate slots in an LKB lexical entry
ba_1 := v-intr & [STEM <"ba">, PHON <"ba">, ENGL-GLOSS <"come">, SYNSEM.LKEYS.KEYREL.PRED "_ba_v-intr_rel", EXAMPLE "E-ba-aa", GLOSS "3S-come-NEG.IMPERF", FREE-TRANSL "he didn't come; he didn't measure up (to a task"].	\lx _ number := \sl1 \sl2 & [STEM <" \lx " >, PHON <" \ph ">, ENGL-GLOSS <" \ge ">, SYNSEM.LKEYS.KEYREL.PRED " _ \lx _ \sl1 \sl2 _rel", EXAMPLE " \xv ", GLOSS " \xg ", FREE-TRANSL " \xe "].

Table 4. Match between Toolbox entry and LKB entry

[13] Dakubu (2009).

4. From the basic resources to the *MultiVal* database

The database itself is a standard relational database.[14] The steps to be described in this section are mapping the grammar specific verb lexicons to a multi-purpose database. In this set of operations, the LKB data is loaded into GA_LEXICON_vol2, CL_LEXICONVOL2, and SRG_LEXICONVOL2, before it is copied into the table MULTI_LEXICON_vol2. The web application reads from the MULTI_LEXICON_vol2 table.

field	GA_LEXICON_vol2	CL_LEXICONVOL2	SRG_LEXICONVOL2	MULTI_LEXICON_vol2
id				x
type	x	x	x	x
orthography	x	x	x	x
language				x
sas	x	x	x	x
fct	x	x	x	x
sit	x	x	x	x
aspect	x	x	x	x
parent	x	x	x	x
phon	x			x
engl_gloss	x			x
example	x			x
example_of_type		x	x	x
gloss	x			x
free_transl	x			x

Table 5. The MultiVal database tables

In creating the MultiVal database, we use information from both the lexical entries of the grammars and the lexical types of these entries. From the relevant lines in the lexical entries, information is directly copied over, whereas in projecting from the lexical types, we use a manually created *conversion list* for each language, *interpreting the type labels as defined in the particular grammar to the values of the discriminants of MultiVal.* We describe below the particulars relative to each grammar.

4.1 From the lexicon of GaGram to MultiVal

We read the relevant GaGram files and store the data in the GA_LEXICON _vol2 table. We update this table from the conversion list for Ga lexical types to MultiVal specifications; an example of an entry in this list (with in total 144 entries, based on types) is given in Table 6 ('*v_suAg-vtrVid*' being a lexical type in Ga, for a type of oblique arguments headed by verbs):

[14] Named *Derby,* an Apache DB project.

v_suAg-vtrVid
SAS: NP + NP + VP
FCT: transwithOblique
SIT: ternaryRel

Table 6. Entry in the Ga conversion list from Ga lexical types to MultiVal discriminant values

Each entry in the list contains a parent type with the fields 'sas', 'fct', and 'sit'. The data is inserted in the following manner, exemplifying from the entry in Table 6: For each row with parent = 'v_suAg-vtrVid', we set SAS = "NP+NP+VP" and FCT = 'intransWithOblique' and SIT = 'ternaryRel'. In other words, we perform a 'conversion' from the lexical type to the specifications in terms of the fields 'sas', 'fct', and 'sit'.

We next move the data from the GA_LEXICON_vol2 table to the MULTI_LEXICON_vol2 table. The Ga language has IDs in the series 200.000-300.000, and the language field has the value "ga".

Content-wise, it may be noted that types with '*sv-*' were not given any value other than 'SVC' (for 'serial verb construction') in the conversion list, for the reason that in an SVC there are as many verb-headed argument frames as there are verbs, and for reasons of time we at present have not decided on conversion keys for such cases.

4.2 From the lexicon of SRG to MultiVal

We read the files: letypes.tdl and lexicon.tdl in SRG and store the data in the table SRG_LEXICONVOL2. A 'conversion' list like in the previous case (with in total 215 types) is used to update the SRG_LEXICONVOL2 table. An example from the Spanish conversion list ('_le' for 'lexical entry'):

v_-_nsbj_le
SAS: +
FCT: intransImpers
SIT: weatherProcess
Example of type: llueve[15]

Table 7. Entry in the Spanish conversion list

Each entry in the list contains a parent type with the fields 'sas', 'fct', 'sit', and 'example of type'. For each row in table SRG_LEXICONVOL2 where parent = 'v_-_nsbj_le', we set sas = '+' and fct = 'intransImpers' and sit = 'weatherProcess' and 'example of type' = 'llueve'. We update the MULTI_LEXICON_vol2 table from SRG_LEXICONVOL2 table. The Spanish grammar has IDs in the series 300.000-400.000, and the 'language' field is set to: "sp".

The '+' in the case illustrated indicates an empty subject, necessary in impersonal constructions. Reflecting a formal feature of the verb type system, the entry below exemplifies how a type encoding *optionality*, represented by '*', is converted:

v_pp*_dir-prn_le
SAS: NP+PPdir+NPrefl NP+NPrefl
FCT: intransReflxWithOptDirectional
SIT: Example: se extiende (hacia el sur)[16]

Table 8. Optionality in the Spanish conversion list

Since any SAS value in the search interface is a unique frame, the two individual frames subsumed by the optionality marking, viz. 'NP+PPdir+NPrefl' and 'NP+NPrefl', need to be retrieved independently. To this end they are entered conjoined as value of 'SAS' in the conversion rule (see Table 8), and in the search interface they function independently, but give the same lexical verb-id as result, i.e., *ir* 'go' with the id 'vprn-pp_dir'.

4.3 From the lexicon of Norsource to MultiVal

We read the NorSource files lex2.open.tdl, lex1.close.tdl, lex4.v-lrg.tdl, norsk.tdl, and lex-types-v.tdl, and store the data in table CL_LEXICONVOL2. We update this table from a conversion list like in the previous cases (for Norsource with in total 348 entries). An example from thi list is given in Table 9:

v-intrImpersPrtcl
SAS: "EXPL+adpos"
FCT: intransImpersonalWithParticle
SIT: weatherProcess
Example of type: det klarner opp[17]

Table 9. Entry in the Norwegian conversion list

Each entry in the list contains a parent type with the fields 'sas', 'fct', 'sit', and 'example'. For each row in table CL_LEXICONVOL2 where parent = 'v-intrImpersPrtcl', we set sas = "EXPL+adpos" and fct = 'intransImpersonalWithParticle' and sit = 'weatherProcess' and example = 'det klarner opp'. We updated table MULTI_LEXICON_vol2 from table CL_LEXICONVOL2. The Norwegian grammar has IDs in the series 100.000 – 200.000.

5. Extendability

We assume that the lexicon of any grammar of the type considered can in principle be mapped onto MultiVal; inclusion of further such lexicons is under consideration. It is also likely that procedures can be defined for other lexicalistic grammar frameworks, although this has not yet been attempted.

The goal of having broad aligned valence repositories however will necessitate further strategies as well. One type of strategy could be that an interconnection between existing mono- and multilingual repositories be established through a common communication script rather than through actual inclusion of one system in

[15] 'It rains'. The entry itself so far has no translation.

[16] 'It spreads (towards de south)'. The entry itself so far has no translation.

[17] 'It clears up'. The entry itself so far has no translation.

another. Still another strategy should be considered which not only exploits existing resources, but opens for the expansion of existing resources, or creation of new resources in ways guaranteeing alignment with existing resources. Such expansion or creation could be done either manually, or through automatic or semi-automatic induction from corpora. Below we describe an existing valence annotation system based on 'manual' interaction, and reflect on how such a system could be made interoperable with MultiVal.

TypeCraft (Beermann and Mihaylov 2013) is a linguistic service featuring a multi-lingual database and an online Interlinear Glosser which in addition to morpheme and word level annotations allows phrase level tagging.

Figure 2 shows a Ga IGT seen from inside of the TypeCraft (TC) linguistic editor. The Editor uses the standard tier format for interlinear glossing. In addition, Phrase level annotation, here called Construction Labeling, can be added through the use of an additional annotation matrix, shown below the IGT. The rightmost part of the screenshot, furthermore, shows drop-down menus for 8 named phrasal parameters.

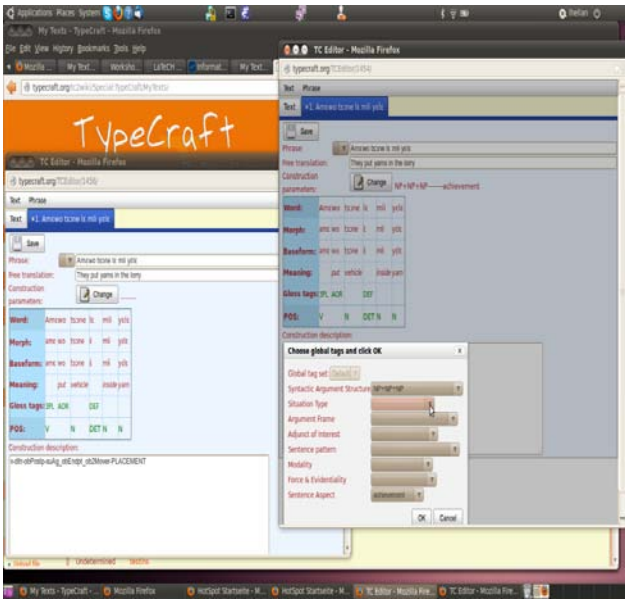


Figure 2: Valence-related annotation in TypeCraft

In this way TC allows the harvesting of valence information in a linguistic environment designed for the manual annotation of data, especially from lesser described languages. TC 2.0 which is at present under development will allow the import of data from other linguistic platforms (Bouda & Beermann) directly into TypeCraft, which then can be used as a tool that allows the easy addition of valence annotation to already structured data.

TC uses its own phrasal-level tagset. It therefore would be essential for MultiVal to support the conversion of TC phrase level annotations to MultiVal discriminants. Creation of valence banks from information stored in deep grammars, such as the HPSG suite of grammars discussed here, depended on the existence of fairly extensive well-curated grammars. And while grammar development is resource extensive, and in most cases requires the work of well-funded projects, this is not the case for data harvested through TC.

6. Outlook

Essential to the MultiVal enterprise is that it is 'montonic': the content of each provenance resource is fully respected and subjected to no changes, it is only in the projection onto the common format that 'harmonizations' take place, and then with specifications essentially neutral relative to those of the inputs.

This does not mean that one has found the key to creating an all-encompassing repository of verb valence: even the notions and codes used in MultiVal have their tradition-dependencies, and for the building of multilingual valence repositories at large, the main challenges may well remain the linguistic ones. Throughout classical and modern linguistics, one has rarely managed to design valence systems that could be applied across languages; and alignment between valence systems as such remains even more difficult, as theories or conventions tend to be geared to different parameters of specification, without there being a defined 'outer grid' against which all specification parameters could be mapped. In principle the construction of such an 'outer grid' should not be impossible, but a complicating factor is the circumstance that a reliable *semantic* space of description is yet to be constructed, accommodating situation types and roles. This is the research area that perhaps most of all requires progress in order for a resource like MultiVal to come into full operation.

7 References

Beermann, D. and Mihaylov, P. (2013). Collaborative databasing and Resource sharing for Linguists. *Languages Resources and Evaluation*. Springer, 1-23.

Bender, E. M., Drellishak, S., Fokkens, A., Poulson, L. and Saleem, S. (2010). Grammar Customization. *Research on Language & Computation*, Volume 8, Number 1, 23-72.

Bender, E., Schikowski, R., and Bickel, B. (2012) Deriving a Lexicon for a Precision Grammar from Language Documentation Resources: A Case Study of Chintang. *Proceedings of COLING 2012*, pp. 247-262.

Bouda, P. and Beermann, D. (2014). Implementing Annotation Graphs for Advanced Convertibility of IGT data. CCURL Workshop, LREC 2014

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, 2009. *Ga-English Dictionary with English-Ga Index.* Accra: Black Mask Publishers

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*. CSLI Publications, Stanford. (/http://csli-publications.stanford.edu/)

Hellan, L., Johnsen, L. And Pitz, A. (1989) TROLL. Ms, NTNU.

Hellan, L. (2007). On 'Deep Evaluation' for Individual Computational Grammars and for Cross-Framework Comparison. In: T.H. King and E. M. Bender (eds) *Proceedings of the GEAF 2007 Workshop*. CSLI Studies in Computational Linguistics ONLINE. CSLI Publications. *http://csli-publications.stanford.edu/*

Hellan, L. (2008). Enumerating Verb Constructions Cross-linguistically. In *Proceedings from COLING 2008 Workshop on Grammar Engineering Across frameworks*. Manchester.

Hellan, L. and Dakubu, M.E.K. (2009): A methodology for enhancing argument structure specification. In *Proceedings from the 4th Language Technology Conference (LTC 2009)*, Poznan.

Hellan, L. and Dakubu, M.E.K. (2010). *Identifying Verb Constructions Cross-linguistically*. SLAVOB series 6.3, Univ. of Ghana.

Hellan and Bruland (2013). Constructing a Multilingual Database of Verb Valence. Paper presented at *NoDaLiDa* 2013.

Hellan, L., Beermann, D., and Bruland, T. (2013). A multilingual valence database for less resourced languages. In *Proceedings from the 6th Language Technology Conference (LTC 2013)*, Poznan.

Hirzel, H. (2006). Deriving LKB lexicons from Toolbox. Talk given at Workshop on Grammar Engineering, NTNU, June 2006.

Marimon, M. (2013). The Spanish DELPH-IN Grammar. *Language Resources and Evaluation*, 47(2), 371-397.

Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. Chicago University Press.

Sag, I., Wasow, T. and Bender, E. (2003). *Syntactic Theory*. CSLI Publications, Stanford.