A Parametric Approach to Implemented Analyses: Valence-changing Morphology in the LinGO Grammar Matrix

Christian Curtis
University of Washington
Department of Linguistics
cmc3c@uw.edu

Abstract

I describe an analysis of valence-changing verbal morphology implemented as a library extending the LinGO Grammar Matrix customization system. This analysis is based on decomposition of these operations into rule components, which in turn are expressed as lexical rule supertypes that implement specific, isolatable constraints. I also show how common variations of these constraints can be abstracted and parameterized by their axes of variation. I then demonstrate how these constraints can be recomposed in various combinations to provide broad coverage of the typological variation of valence change found in the world's languages. I evaluate the coverage of this library on five held-out world languages that exhibit these phenomena, achieving 79% coverage and 2% overgeneration.

1 Introduction

The LinGO Grammar Matrix (Bender et al., 2002) is a resource, rooted in the Head-driven Phrase Structure Grammar (HPSG) formalism (Pollard and Sag, 1994), that enables linguists to create implemented precision grammars. The core of the Grammar Matrix is a collection of types and constraints expected to be cross-linguistically useful, such as lexical and phrase rule types, feature geometry, and types implementing compositionality and long-distance dependency resolution. These analyses embed linguistic knowledge developed and tested by linguists and grammar writers over many years, in implementations of grammars at both large and small scales, in a framework that provides infrastructure and context for reuse in development of new grammars. Beyond reuse and rapid development of new grammars, aspects of the *engineering* purpose of the Grammar Matrix, the Matrix also serves two scientific goals, as articulated by Bender et al. (2010): first, to support linguistic hypothesis testing through grammar engineering; and, second, to combine both breadth of typological research and depth of syntactic analysis into a single computational resource.

In this work I present an analysis of valence-changing verbal morphology in order to test two primary hypotheses: first, that a typologically-informed set of implemented valence-changing operations can cover a meaningful proportion of the incidence of valence change in the world's languages; and, second, that these valence-changing operations can be implemented in a "building-block" fashion by building up complete valence change operations from isolated, common elements that can be reused and recombined in varying combinations. In order to test these hypotheses, I developed a library for valence-changing verbal morphology for the Grammar Matrix customization system, and evaluated its performance when modeling valence change from five held-out languages from different familial and areal groups.

2 Relevant elements of HPSG and the Grammar Matrix

The foundation of HPSG is the representation of each linguistic sign as a typed feature structure. This typed feature structure (Carpenter, 1992) is a structured object consisting of defined attributes, or features, the values of which are themselves other typed feature structures. The type of a feature structure determines which features appear in it. Thus, the entire feature structure of a sign forms a directed graph, where each node is a feature structure, and each edge is labeled with a feature name. Each node in the graph can be reached by following a sequence of labeled edges from the root. However, this path need not be unique; two (or more) paths through different feature structures may reach the same node.

Proceedings of the 2nd Int. Workshop on Resources and Tools for Derivational Morphology (DeriMo 2019), pages 111–120,
Prague, Czechia, 19-20 September 2019.

Disjoint feature structures can be combined, or *unified*, where their graphs (or a graph and a subgraph) are isomorphic. Unification is constrained such that the types of each node must be compatible; that is, that (a) the features present at the node must be valid for both node types, and (b) the types of each value the nodes have in common are also compatible. In this way the constraints expressed by each feature structure are satisfied. Note that this definition is recursive: each node must unify as well as each descendant node.

A grammar in this paradigm is comprised of the following main elements:

- **lexical types**, constraints inherited by words in the lexicon;
- **lexical rule types**, constraints on how stems give rise to inflected and derived forms;
- **grammar rule types**, constraints on how words combine into phrases, and how phrases combine;
- **foundational types**, types that constrain feature values (*e.g.*valid values of the CASE feature); and
- **instances**, instantiations of lexical types and lexical and grammar rule types.

This brief description illustrates two distinctive attributes of HPSG,[1] as described in Sag et al. 2003, Chapter 9. First, grammars are based on constraint satisfaction through unification (as contrasted with a transformational approach). Second, the grammar's view of syntax is strongly lexical: constraints originate with instances of lexical types and a distinction is made between word-internal rules and syntactic rules, the latter having no access to the former.

The Grammar Matrix implements a restricted formalism, described in Copestake (2002) and referred to as the DELPH-IN joint reference formalism, that significantly limits the available operations on feature structures. For example, the Joint Reference Formalism disallows set-valued features and relational constraints, and all structures must be acyclic.

2.1 Valence and argument structure

In the revised[2] conception of HPSG, the valence of a particular sign—that is, the specification of what other signs it must combine with to become saturated—is conveyed by the SUBCAT feature, which is defined as the append of the SUBJ, SPR, and COMPS lists (representing, respectively, the subject, specifier, and complements of the sign) (Pollard and Sag, 1994, p. 375). Subsequently, Manning and Sag (1998) proposed a modification whereby SUBCAT became a means to express the argument structure of a lexical sign, and specifically as a distinct entity from the SUBJ, SPR, and COMPS valence lists. The SUBCAT feature was renamed to ARG-ST to indicate this revised role.

This separation and its concomitant materialization of the mechanisms for linking argument structure and valence lists, making them available for manipulation, is essential to the implementation of valence change in this work. As I describe in more detail below, operations such as the passive rely on changing the relationship between syntactic and semantic roles played by a verb's arguments.

2.1.1 Grammar Matrix customization system

The Grammar Matrix customization system (Bender et al., 2010) combines a structured means of eliciting typological characteristics, validating responses for consistency, and using those choices to combine Matrix core grammar elements with stored analyses of various linguistic phenomena into a customized grammar. These stored analyses can include both static representations of cross-linguistically common phenomena as well as dynamically-generated implementations that embody language-specific variations. Elicitation is accomplished via a dynamic, iteratively-generated HTML questionnaire, which records the responses (while validating the consistency of both individual responses and their combination) in a structured choices file. This choices file is then processed by the customization script to produce the customized grammar. The system components and their relationships are shown in Figure 1 (from Bender et al., 2010, p. 31).

The stored analyses of linguistic phenomena in the customization system are organized into conceptual "libraries." These libraries also provide elements of the questionnaire, customization routines, and validation logic associated with the phenomena analyses they control. Representative libraries include word order (Fokkens, 2010), sentential negation (Crowgey, 2012), argument optionality (Saleem, 2010;

[1]These attributes also apply to other grammar approaches in the same tradition.

[2]With respect to earlier chapters of Pollard and Sag 1994.

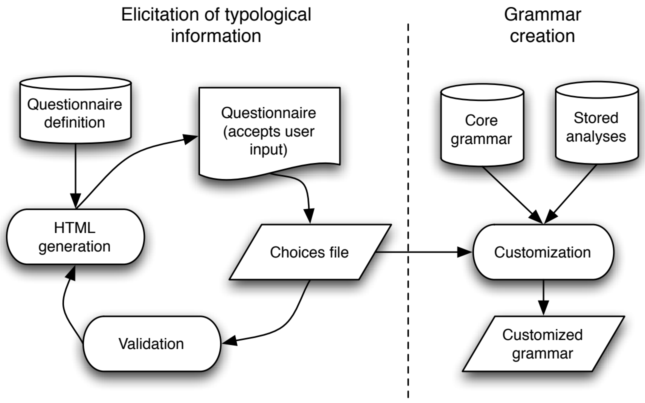

Figure 1: Customization system overview

Saleem and Bender, 2010), and information structure (Song, 2014), among others. Libraries may also interact and depend on facilities provided by other libraries; libraries such as the one presented here that implicate morphology may have relatively tighter coupling to the morphotactics library, for example.

2.2 Morphotactics

The Grammar Matrix customization system includes mechanisms for implementing morphosyntax, including the obligatoriness, ordering, and co-occurrence of position classes, and the definition and instantiation of lexical rules to implement inflectional (and, to a limited degree, derivational) morphology. The original morphotactics library was developed by O'Hara (2008), with argument optionality added by Saleem (2010). The current morphotactics framework is the result of significant modification and improvement by Goodman (2013).

2.3 Minimal Recursion Semantics

The Grammar Matrix uses Minimal Recursion Semantics (MRS; Copestake et al. 2005) as its semantic representation, integrated into its HPSG mechanisms and feature structures. In MRS, the primary unit of interest for semantics is the elementary predication (EP), which is a single relation and its arguments, identified by a label. EPs are never embedded in other EPs, but are instead grouped as flat elements in a bag.[3] Typical lexical items contribute a single EP; phrase structure rules construct a bag of EPs by appending the bags of EPs of all the phrase daughters and may contribute EPs themselves. This flat representation is underspecified as to scope, so an additional set of constraints are applied that define a restricted set of ways in which EPs can be related via scope relations. Scopal arguments are expressed via handle relationships, in which a handle is equal, modulo quantifiers, to a label. This relationship, denoted as *qeq* or $=_q$, allows semantic composition to be defined simply while preserving the scope constraints that could contain intervening quantifiers. This distinction between scopal and non-scopal relationships is essential to expressing certain valence-changing phenomena such as the causative.

3 Typology of valence change

Valence, by analogy to the valence of atoms in chemistry (Tesnière, 1959), refers to the number of core syntactic arguments a verb in a given clause type takes. All human languages have both intransitive and transitive clauses: intransitive clauses have a single argument, the subject (denoted S); transitive clauses have two arguments, the transitive subject (denoted A) and the transitive object (denoted O) (Dixon, 1979).

Many languages permit verbal derivations that alter the argument structure of verbs, either increasing or decreasing the valence and changing the relationship of realized arguments to syntactic roles. In analyzing the cross-linguistic range of these operations below, I follow the broad conceptual framework provided by Haspelmath and Müller-Bardey (2004) (henceforth H&MB) and group the operations first by whether

[3] In a bag, unlike in a set, EPs may be repeated.

they reduce or increase valence, and second by whether they affect the subject or object. I also retain their focus on verbal valence-changing morphology (thus excluding, e.g., periphrastic constructions).

3.1 Valence-reducing operations

The primary types of subject-removing operation to consider are the anticausative and the passive. Both remove the subject (A) and move the former object (O) into the subject position; the essential distinction between them is that the anticausative removes the A argument entirely, while the passive merely moves it to the periphery (H&MB). The Turkish [tur] anticausative and passive in Mam [mam] (Mayan family), are illustrated in (1) and (2), respectively:

(1) a. *Anne-m kapı-yı aç-tı*
mother-1SG door-ACC open-PAST(3SG)
'My mother opened the door.' [tur]

b. *Kapı aç-tı-dı*
door open-ANTIC-PAST(3SG)
'The door opened.' [tur] (H&MB, p. 5)

(2) a. *ma ch-ok t-b'iyo-'n Cheep kab' xjaa*
PAST 3PL+O-DIRectional 3SG+A-hit-DIR José two person
'José hit two people.' [mam]

b. *ma chi b'iy-eet kab' xjaa (t-u'n Cheep)*
PAST 3PL+S hit-PASS two person 3SG-REL/AGENT José
'Two people were hit (by José).' [mam]
(England, 1983, in Dixon and Aikhenvald, 1997, p. 75)

Analogous to the anticausative, the object-removing operation where the object O is completely removed is referred to as the deobjective (H&MB) or the absolutive antipassive (Dayley, 1989, as cited in H&MB). A related form, the "potential deobjective," expresses disposition of an agent rather than a real action; however, this semantic distinction is not relevant to this analysis.

(3) a. *Sake a-ku*
sake 1SG.TR-drink
'I drink sake.' [ain]

b. *I-ku-an*
DEOBJ-drink-1SG.INTR
'I drink.' [ain] (Shibatani, 1990, in H&MB, p. 3)

The deaccusative (H&MB) or antipassive[4] (Dixon and Aikhenvald, 2000) is similar, but instead of completely removing the underlying O argument, moves it out of the core to the periphery, as illustrated by the Hungarian [hun] deaccusative in (4).

(4) a. *Az orvos szán-ja a beteg-et*
the doctor pity-3SG the patient-ACC
'The doctor pities the patient.' [hun]

b. *Az orvos szán-akoz-ik a beteg-en*
the doctor pity-DEACC-3SG the patient-SUPERESS
'The doctor feels pity for the patient.' [hun] (Károly, 1982, in H&MB, p. 4)

3.2 Valence-increasing operations

3.2.1 Subject-adding

Cross-linguistically the most common valence-changing category (Bybee, 1985), the causative adds a new subject (A), the causer of the event described by the verb. The addition of a causer to an intransitive

[4] Dixon and Aikhenvald use 'antipassive' to refer to both constructions, noting simply that "the [underlying O] argument may be omitted" (p. 9)

verb can simply move the underlying subject (S) into an object (O) position, as illustrated by the Vengo [bav] (Grassfields Bantu) causative in (5):

(5) a. *nw nìi táa nìì*
he enter in house
'He entered the house.' [bav]

b. *m nìi-s nw táa nìì*
I enter-CAUS him in house
'I made him enter the house.' [bav] (Schaub, 1982, in H&MB, p. 11)

The situation with underlying transitive verbs is more complex, as there are different strategies for dealing with the underlying subject (causee), given the presence of an already-existing direct object (O). H&MB identify three such strategies, illustrated in (6): (6a) causee as an indirect object, as in Georgian [kat], (6b) causee as instrumental phrase, as in Kannada [kan], and (6c) causee as second direct object, as in Imbabura Kwicha [qvi].

(6) a. *Mama-m Mzia-s daanteb-in-a cecxli*
father-ERG Mzia-DAT light-CAUS-AOR:3SG fire(ABS)
'Father made Mzia light the fire.' [kat] (Harris, 1981, in H&MB, p. 12)

b. *Raamanu manga-gal-inda Siite-yannu huduki-si-danu*
Rama(NOM) monkey-PL-INSTR Sita-ACC search-CAUS-3SG
'Rama had the monkeys search for Sita.' [kan] (Cole and Sridhar, 1977, in H&MB, p. 12)

c. *Juzi-ka Juan-ta ruwana-ta awa-chi-rka*
José Juan-ACC poncho-ACC weave-CAUS-3SG
'José made Juan weave a poncho.' [qvi] (Cole, 1982, in H&MB, p. 12)

Other subject-adding constructions are structurally similar to the causative, such as the affective ('indirect passive') in Japanese [jpn]. A crucial aspect of the causative and similar constructions is the addition of a new EP which functions as a scopal operator with respect to the verb's own EP and takes as an argument the added participant. This is distinguished from the applicative (below), which is non-scopal and does not affect semantic roles.

3.2.2 Object-adding

Object-adding constructions can collectively be grouped under the term 'applicative,' which subsumes a broad variation in potential roles for the added structural argument. The prototypical applicative is the benefactive, as demonstrated in the Indonesian [ind] alternation in (7). In many languages (e.g., in the Bantu family, especially) applicatives can serve many other functions, including possessor-raising, instrumental, and locative applicatives.

(7) a. *Orang itu masak ikan untuk perempuan itu*
man DEF cook fish for woman DEF
'The man cooked fish for the woman.' [ind]

b. *Orang itu memasakan perempuan itu ikan*
Orang itu me-masak-kan perempuan itu ikan
man DEF TR-cook-BEN woman DEF fish
'The man cooked the woman fish.' [ind] (Chung, 1976, p. 58)

(8) a. *Ali memi telefisi untuk ibu-nja*
Ali TR.buy television for mother-his
'Ali bought a television for his mother.' [ind]

b. *Ali mem-beli-kan ibu-nja telefisi*
Ali TR-buy-APPL mother-his television
'Ali bought his mother a television.' [ind] (Chung, 1976, in Wunderlich, 2015, p. 21)

4 Analysis

The overall approach I followed was to decompose the high-level, linguistically-significant valence-changing operations into their component operations on feature structures. These individual component operations can then be selected by the customization system and composed to achieve the high-level result. The components I selected to analyze and implement included addition and removal of subjects and objects, case constraints and alternations, and argument reordering. For the purpose of illustration, I focus here on object- and subject-adding operations.

4.1 Object addition

In adding an argument, there are several underlying operations in my analysis: (a) adding an argument to the COMPS list;[5] (b) constraining the added argument (or promoted subject), e.g. to be an NP or PP (HEAD *noun* or *adp*), or applying a CASE constraint; (c) appending the new argument's non-local dependencies to the rule mother's list;.[6] (d) contributing an added elementary predication (EP) via C-CONT; (e) linking the new EP's ARG1 to the daughter's INDEX; and (f) linking the new EP's ARG2 to the new argument's INDEX.

The addition of a new EP to the rule output is not as straightforward and requires some additional discussion. To motivate this analysis, consider the example of the benefactive from Indonesian in (8). In this example, the addition of the benefactive applicative suffix *-kan* in (7b) adds an argument position to the verb, which is filled by *perempuan itu* "the woman."

Notionally, the benefactive is adding a third semantic argument to the verb, which would add a hypothetical ARG3 to the EP contributed by the verb; however, this would seem to violate the principles of semantic composition in Copestake et al. (2005), namely, that composition consists solely of *concatenation* of daughter RELS values, not modification. More concretely, there is no EP-modifying operation available within the algebra of Copestake et al. (2001).

The solution is to have the lexical rule contribute a new EP, which takes both the EP contributed by the verb and the additional syntactic argument as semantic arguments. The predicate value for this new EP will provide the particular species of applicative (e.g., benefactive, as here). This new EP contributes its own event and takes as its arguments the respective indexes of the input and the added argument. In this analysis I treat the added arguments as non-scopal, with no intervening handle relationships; this contrasts with my analysis of subject addition below. The MRS resulting from this analysis is shown below in (9):

(9)
$$\left[\text{RELS}\ \left\langle \begin{bmatrix}\textbf{\_memi\_v\_buy} & \\ \text{ARG0} & \boxed{4}\ \textit{event} \\ \text{ARG1} & \boxed{1} \\ \text{ARG2} & \boxed{2}\end{bmatrix}, \begin{bmatrix}\textbf{named} & \\ \text{ARG0} & \boxed{1}\end{bmatrix}, \begin{bmatrix}\textbf{\_telefisi\_n\_TV} & \\ \text{ARG0} & \boxed{2}\end{bmatrix}, \begin{bmatrix}\textbf{\_ibu\_n\_mother} & \\ \text{ARG0} & \boxed{3}\end{bmatrix}, \begin{bmatrix}\textbf{benefactive} & \\ \text{ARG0} & \textit{event} \\ \text{ARG1} & \boxed{4} \\ \text{ARG2} & \boxed{3}\end{bmatrix} \right\rangle \right]$$

With all these elements combined, a complete rule implementing the benefactive can be implemented as illustrated in (10). This rule, however, in combining the distinct operations identified above, obscures common elements that can be reused for other similar object-adding operations. Reviewing the operations, it is evident that they vary along different axes, as summarized in Table 1.

This leads to a simplification and optimization: in the same way that the intransitive and transitive forms of subject removal can be viewed as variants of a single abstract analysis along the transitivity axis, these building-block operations can also be treated as being parameterized along their axes of variation and then combined to make the final rule type.

[5]Note that, cross-linguistically, the added argument can added either more- or less-obliquely to the verb's existing dependencies (i.e., at the head or tail of the COMPS list).

[6]To conserve space, NON-LOCAL features are omitted from the examples presented.

(10)
$$\begin{bmatrix} \textit{benefactive-lex-rule} & \\ \text{SYNSEM | LOCAL | CAT | VAL | COMPS} & \left\langle \boxed{1}, \begin{bmatrix} \text{LOCAL} & \begin{bmatrix} \text{CAT} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \, \rangle \\ \text{COMPS} & \langle \, \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \text{CONT | HOOK | INDEX} & \boxed{2} \end{bmatrix} \right\rangle \\ \text{C-CONT} & \begin{bmatrix} \text{RELS} & \left\langle ! \begin{bmatrix} \textit{event-relation} & \\ \text{PRED} & \textit{``benefactive''} \\ \text{ARG1} & \boxed{6} \\ \text{ARG2} & \boxed{2} \end{bmatrix} ! \right\rangle \\ \text{HCONS} & \langle ! \quad ! \rangle \end{bmatrix} \\ \text{DTR} & \begin{bmatrix} \textit{verb-lex} & \\ \text{SYNSEM | LOCAL} & \begin{bmatrix} \text{CAT | VAL | COMPS} & \boxed{1} \\ \text{CONT | HOOK | INDEX} & \boxed{6} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Concretely, taking these operations in turn, the first operation (adding the argument) needs to have variants for adding an argument: (a) to intransitive or transitive verbs; and (b) at the front or end of the COMPS list. That is, the lexical rule type implementing each of the component operations can be viewed as the output of a function: $f : tr \in \{intrans, trans\} \times pos \in \{front, end\} \rightarrow lrt$.

To illustrate this variation, the rule type at (11) adds an argument to the (empty) COMPS list for an intransitive verb, and the rule at (12) adds an argument at the front of the COMPS list for a transitive verb and links the INDEX of that argument to its second semantic argument (ARG2).

(11)
$$\begin{bmatrix} \textit{added-arg2of2-lex-rule} & \\ \text{SYNSEM | LOCAL | CAT | VAL | COMPS} & \left\langle \begin{bmatrix} \text{LOCAL} & \begin{bmatrix} \text{CAT | VAL} & \begin{bmatrix} \text{SPR} & \langle \, \rangle \\ \text{COMPS} & \langle \, \rangle \end{bmatrix} \\ \text{CONT | HOOK | INDEX} & \boxed{1} \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{C-CONT | RELS} & \left\langle ! \begin{bmatrix} \text{ARG2} & \boxed{1} \end{bmatrix} ! \right\rangle \\ \text{DTR | SYNSEM | LOCAL | CAT | VAL | COMPS} & \langle \, \rangle \end{bmatrix}$$

(12)
$$\begin{bmatrix} \textit{added-arg2of3-lex-rule} & \\ \text{SYNSEM | LOCAL | CAT | VAL | COMPS} & \left\langle \begin{bmatrix} \text{LOCAL} & \begin{bmatrix} \text{CAT | VAL} & \begin{bmatrix} \text{SPR} & \langle \rangle \\ \text{COMPS} & \langle \rangle \end{bmatrix} \\ \text{CONT | HOOK | INDEX} & \boxed{1} \end{bmatrix} \end{bmatrix}, \boxed{2} \right\rangle \\ \text{C-CONT | RELS} & \left\langle ! \begin{bmatrix} \text{ARG2} & \boxed{1} \end{bmatrix} ! \right\rangle \\ \text{DTR | SYNSEM | LOCAL | CAT | VAL | COMPS} & \langle \boxed{2} \rangle \end{bmatrix}$$

rule component	**varies by**
added argument	position (obliqueness), number of existing args
constraint on new argument	position (obliqueness), constraint (*e.g.* case, head)
non-local dependencies	position (obliqueness)
new EP's PRED value	predicate
new EP's ARG1	does not vary
new EP's ARG2	position (obliqueness)

Table 1: Rule component axes of variation (benefactive)

The remaining operation components can likewise be separated into independent rule types, isolated to a particular element and parameterized on its axis of variation. These "building blocks," as rule component supertypes, can then be assembled as inherited constraints on a complete applicative rule type, ready to be instantiated in a grammar. The partial inheritance tree showing these rule component supertypes for the notional benefactive full rule type described here is illustrated in Figure 2.

4.2 Subject addition

The canonical subject-adding operation is the causative, which introduces a new argument into the subject role and moves the erstwhile subject into another position. In contrast to the applicative, I treat the causative as a scopal predicate: the "causing" EP outscopes the underlying verb's EP and so provides the HOOK feature values for the entire VP.

Consistent with the strategy in Copestake et al. (2001), the scopal relationship is expressed by a handle constraint (HCONS) rather than directly, representing equality modulo quantifiers ($=_q$).

Similarly to my analysis of the applicative, the causative can also be decomposed into component operations, again parameterized along the axes of cross-linguistic variation.

5 Implementation in the Grammar Matrix

The Grammar Matrix customization system (Bender et al., 2010) combines a structured means of eliciting typological characteristics, validating responses for consistency, and using those choices to combine Matrix core grammar elements with stored analyses of various linguistic phenomena into a customized grammar. These stored analyses can include both static representations of cross-linguistically common phenomena as well as dynamically-generated implementations that embody language-specific variations.

My implementation of a library leverages the existing morphotactics machinery in the customization system (Goodman, 2013) by adding options to the questionnaire for grammar writers to attach valence-changing operations to lexical rule types, along with the relevant parameters (e.g., position of erstwhile subject) necessary to generate the operations. My extensions to the grammar customization scripts, in turn, use the selections in the choices file to generate the appropriate parameterized and common rule components, and then combine them into types to be instantiated.

While developing the library, two types of tests were used. Initially, I developed small, abstract pseudo-languages to exercise specific operations and combinations; I then attempted to model valence change in three natural languages, Lakota [lkt], Japanese [jpn], and Zulu [zul], and produced test suites of grammatical and ungrammatical examples. During this phase of development, I continued to revise my analyses and code to achieve full coverage of the examples. Once this phase was complete, I then froze library development and moved to the evaluation phase, described in the next section.

6 Evaluation

To evaluate the library as developed against a representative sample of the world's languages, I selected five held-out languages, from different familial and areal groups, that had not been used during development. Two languages were selected from descriptive articles intentionally held out, and the rest were selected by drawing randomly from a large collection of descriptive grammars, discarding those without valence changing morphology, until sufficient evaluation languages were collected.

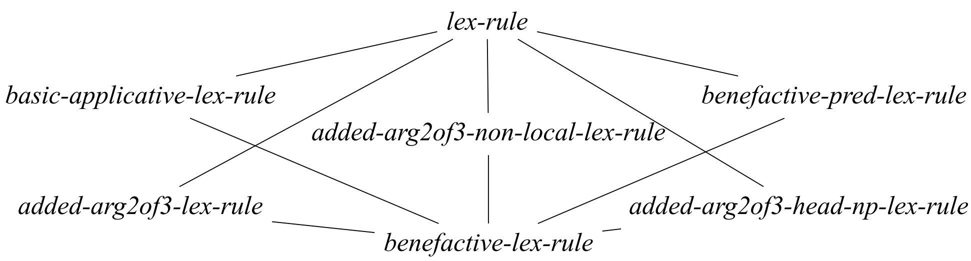

Figure 2: Example of rule component type hierarchy for applicative

I created test suites for each of these languages consisting of grammatical and ungrammatical examples of valence change, and attempted to model the corresponding phenomena using only the facilities available in the customization system questionnaire. I then attempted to parse the test suites using the customization system-generated grammars and recorded which grammatical examples were correctly parsed, which ungrammatical examples were erroneously parsed, and to what extent the parses generated spurious ambiguity. These results are summarized in Table 2.

		examples		performance			
Language	Family	positive	negative	parses	coverage	overgeneration	spurious ambiguity
Tsez [ddo]	NE Caucasian	11	8	10	91%	0%	0%
West Greenlandic [kal]	Eskimo-Aleut	15	14	12	73%	0%	0%
Awa Pit [kwi]	Barbacoan	7	7	5	71%	0%	0%
Rawang [raw]	Sino-Tibetan	11	6	6	55%	0%	0%
Javanese [jav]	Austronesian	13	8	12	92%	13%	0%
Total		57	43	45	79%	2%	0%

Table 2: Test languages test summary and performance

On the test suites for the five held-out languages, this approach as implemented in my library achieved an overall coverage of 79% and an aggregate overgeneration rate of only 2%. The language with the poorest coverage (55%), Rawang [raw], suffered almost entirely due to a relatively rich system of reflexive and middle constructions; my library lacked the ability to fill a valence slot while coindexing with an existing argument and so these examples could not be modeled. The sole example of overgeneration, from Javanese [jav], was similarly due to the inability of the current library to apply a HEAD constraint to an already-existing argument. Neither of these limitations appear to be fundamental, and so modifying the library to include these additional phenomena would be straightforward.

7 Conclusion

In this work I have presented an HPSG analysis of valence-changing verbal morphology, implemented in the LinGO Grammar Matrix, which I evaluated against several held-out languages. The results appear to support the hypothesis that a "building-block" based approach is an effective way to provide significant typological coverage of valence change. By developing and implementing this analysis within the larger Grammar Matrix project, these elements of valence change can be combined and recombined in different ways to test linguistic hypotheses and compare modeling choices, including the interactions of valence change with other phenomena. Although the scope of this work was limited to valence change expressed through verbal morphology, future work might include determining whether this approach can be extended to other phenomena, including, for example, periphrastic valence-changing constructions.

References

Emily M. Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar customization. *Research on Language & Computation* 8(1):23–72.

Emily M. Bender, Dan Flickinger, and Stephan Oepen. 2002. The Grammar Matrix: An Open-Source Starter-Kit for the Rapid Development of Cross-linguistically Consistent Broad-coverage Precision Grammars. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Taipei, Taiwan, pages 8–14.

Joan L Bybee. 1985. *Morphology: A study of the relation between meaning and form*. John Benjamins Publishing, Amsterdam.

Bob Carpenter. 1992. *The logic of typed feature structures*. Cambridge University Press, Cambridge, U.K.

Sandra Chung. 1976. An object-creating rule in Bahasa Indonesia. *Linguistic Inquiry* 7:41–87.

Peter Cole. 1982. *Imbabura Quechua*. Croom Helm, London etc.

Peter Cole and S. N. Sridhar. 1977. Clause union and relational grammar: Evidence from Hebrew and Kannada. *Linguistic Inquiry* 8(4):700–713.

Ann Copestake. 2002. Definitions of typed feature structures. In Stephan Oepen, Dan Flickinger, Jun-ichi Tsujii, and Hans Uszkoreit, editors, *Collaborative Language Engineering*, CSLI Publications, Stanford, CA, pages 227–230.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A. Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation* 3(2):281–332.

Ann Copestake, Alex Lascarides, and Dan Flickinger. 2001. An algebra for semantic construction in constraint-based grammars. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL '01, pages 140–147.

Joshua Crowgey. 2012. *The Syntactic Exponence of Sentential Negation: a model for the LinGO Grammar Matrix*. Master's thesis, University of Washington.

Jon Philip Dayley. 1989. *Tümpisa (Panamint) Shoshone Grammar*, volume 115. Univ of California Press.

R. M. W. Dixon. 1979. Ergativity. *Language* 55(1):59–138.

R. M. W. Dixon and Alexandra Y Aikhenvald. 1997. A typology of argument-determined constructions. In Joan Bybee, John Haiman, and Sandra A. Thompson, editors, *Essays on Language Function and Language Type*, John Benjamins, Amsterdam, pages 71–113.

R. M. W. Dixon and Alexandra Y. Aikhenvald. 2000. *Changing Valency*. Cambridge University Press.

Nora C England. 1983. *A grammar of Mam, a Mayan language*. University of Texas Press, Austin.

Antske S. Fokkens. 2010. Documentation for the Grammar Matrix word order library. Technical report, Saarland University, Saarbrücken.

Michael W. Goodman. 2013. Generation of machine-readable morphological rules from human-readable input. *University of Washington Working Papers in Linguistics* 30.

Alice C. Harris. 1981. *Georgian Syntax: A Study in Relational Grammar*. Cambridge University Press, Cambridge.

Martin Haspelmath and Thomas Müller-Bardey. 2004. Valence change. *Morphology: A handbook on inflection and word formation* 2:1130–1145.

Sándor Károly. 1982. Intransitive-transitive derivational suffixes in Hungarian. In Ferenc Kiefer, editor, *Hungarian Linguistics*, volume 4, pages 185–243.

Christopher D. Manning and Ivan A. Sag. 1998. Argument structure, valence, and binding. *Nordic Journal of Linguistics* 21(2):107–144.

Kelly O'Hara. 2008. *A morphotactic infrastructure for a grammar customization system*. Master's thesis, University of Washington.

Carl Pollard and Ivan A Sag. 1994. *Head-driven Phrase Structure Grammar*. University of Chicago Press.

Ivan A. Sag, Thomas Wasow, and Emily M. Bender. 2003. *Synactic Theory: A Formal Introduction*. CSLI, Stanford, CA, 2nd ed. edition.

Safiyyah Saleem. 2010. *Argument Optionality: A New Library for the Grammar Matrix Customization System*. Master's thesis, University of Washington.

Safiyyah Saleem and Emily M. Bender. 2010. Argument optionality in the LinGO Grammar Matrix. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, Stroudsburg, PA, USA, COLING '10, pages 1068–1076.

Willi Schaub. 1982. *Babungo*. Croom Helm, London etc.

Masayoshi Shibatani. 1990. *The languages of Japan*. Cambridge University Press.

Sanghoun Song. 2014. *A Grammar Library for Information Structure*. Ph.D. thesis, University of Washington.

Lucien Tesnière. 1959. *Eléments de syntaxe structurale*. Librairie C. Klincksieck, Paris.

Dieter Wunderlich. 2015. Valency-changing word-formation. In Peter O. Müller, Ingeborg Ohnheiser, Susan Olsen, and Franz Rainer, editors, *Word-Formation*, De Gruyter Mouton, Berlin/Boston, volume 3, pages 1424–1466.