

©Copyright 2018

Elizabeth Nielsen

# Modeling Adnominal Possession in the LinGO Grammar Matrix

Elizabeth Nielsen

A thesis  
submitted in partial fulfillment of the  
requirements for the degree of

Master of Science

University of Washington

2018

Reading Committee:

Emily M. Bender, Chair

Edith Aldridge

Program Authorized to Offer Degree:  
Linguistics

University of Washington

**Abstract**

Modeling Adnominal Possession  
in the LinGO Grammar Matrix

Elizabeth Nielsen

Chair of the Supervisory Committee:  
Professor Emily M. Bender  
Linguistics Department

This thesis describes the expansion of the LinGO Grammar Matrix customization system with a library handling adnominal possession. This library extends the Grammar Matrix's ability to aid in the rapid prototyping of precision implemented grammars by allowing a typologically widespread phenomenon — adnominal possession — to be modeled within the customization system. This thesis describes an HPSG analysis of adnominal possession that covers most attested adnominal possessive phrase types, and discusses its implementation in the form of a Grammar Matrix library. The library's cross-linguistic generality is evaluated through a series of tests on constructed and natural languages.

## TABLE OF CONTENTS

	Page
Chapter 1: Introduction . . . . .	1
Chapter 2: Background . . . . .	4
2.1 Head-driven Phrase Structure Grammar . . . . .	4
2.2 Minimal Recursion Semantics . . . . .	9
2.3 The LinGO Grammar Matrix . . . . .	12
2.4 Summary . . . . .	14
Chapter 3: Typology of adnominal possession . . . . .	15
3.1 Terminology . . . . .	15
3.2 Semantics of possession . . . . .	16
3.3 Syntactic exponence of possession . . . . .	18
3.4 Summary . . . . .	31
Chapter 4: Analysis . . . . .	32
4.1 Semantic representation . . . . .	33
4.2 Modifier-like possessive constructions . . . . .	34
4.3 Specifier-like possessive constructions . . . . .	53
4.4 Possessor pronouns . . . . .	66
4.5 Agreement . . . . .	71
4.6 Summary . . . . .	74
Chapter 5: Implementation . . . . .	75
5.1 Questionnaire . . . . .	75
5.2 Grammar-producing code . . . . .	79
5.3 Example . . . . .	83
5.4 Summary . . . . .	86

Chapter 6: Evaluation . . . . .	88
6.1 Pseudo-languages . . . . .	89
6.2 Illustrative languages . . . . .	92
6.3 Held-out languages . . . . .	100
6.4 Post-evaluation changes . . . . .	110
6.5 Summary . . . . .	111
Chapter 7: Conclusion . . . . .	112
Bibliography . . . . .	114

## ACKNOWLEDGMENTS

I owe many thanks to all my fellow Grammar Matrix developers for offering advice, encouragement, and the occasional dire warning. Thanks in particular to Michael Haeger, T.J. Trimble, Olga Zamaraeva, Laurie Dermer, Michael Goodman, and Kristen Howell for lending me a hand when I needed it.

I am also grateful to my reader, Edith Aldridge, for providing insight and expertise that filled in my own blind spots.

Finally, many thanks to my adviser, Emily Bender, for helping me through what has been an immensely enjoyable process. I chose to leave my comfort zone and complete a syntax-focused thesis largely based on my suspicion that Emily would be a great adviser to work with; I was not wrong. I am grateful for her clarity, insight, and patience, as she reminded me for the hundredth time that I only needed to model languages that actually existed. Over the past year, I've been introduced by her to an endless source of entertaining puzzles that I hope to be working on for years to come.

## Chapter 1

# INTRODUCTION

This thesis describes an extension of the LinGO Grammar Matrix customization system to cover adnominal possessive constructions. The Grammar Matrix (Bender et al., 2002, 2010) is a system that aids in the rapid production of implemented grammars in the Head-driven Phrase Structure Grammar (HPSG) formalism (Pollard and Sag, 1994). It consists of an online questionnaire, which allows the system to take input describing a given language, and a set of grammatical type declarations that can be combined and modified to build a grammar to the user’s specification. The Grammar Matrix originally covered a more limited range of phenomena, with the intent that it be extended over time. In the past, extensions have taken the form of libraries that cover such phenomena as adjectives (Trimble, 2014) and evidentiality (Haeger, 2017), and other topics. This thesis represents an effort to do the same for adnominal possessive phrases.

This thesis begins with a discussion of some background information necessary to understand the functioning of the Grammar Matrix, including its theoretical underpinnings, found in chapter 2, before discussing the actual extension to the customization system undertaken. Extending the Grammar Matrix customization system with a new library is an exercise in multilingual grammar engineering, which begins with outlining the breadth of typological variation that is attested in the world’s languages. A typological overview of adnominal possession is given here in chapter 3. Next, a syntactic analysis is devised within the HPSG framework that accounts for these phenomena, outlined here in chapter 4. This analysis is then translated into a computer-readable format, known as Type Description Language (TDL) (Copestake, 2002), and the customization questionnaire and back-end Python code are extended so as to build the desired TDL, as detailed in chapter 5. Finally, the library

is tested in three stages: first on invented pseudo-languages, then on a set of illustrative languages which are available during development, and last on a set of held-out languages that are not known until development has ended. Testing is discussed in chapter 6.

A library covering adnominal possessive constructions is of interest because of the widespread occurrence of these constructions in the world’s languages. As noted by Heine (1997), “[p]ossession is a universal domain, that is, any human language can be expected to have conventionalized expressions for it.” Since possessive expressions are so universal, adding coverage for a subset of conventionalized expressions of possession to the Grammar Matrix customization system increases coverage in almost every language. In addition, many languages have two or more distinct types of adnominal possessive constructions (Nichols and Bickel, 2013), which makes the problem of modeling possession in a given language a complex and interesting one, since issues arising from the interaction of these constructions must be considered.

Before discussing the construction of this library, it is worth noting that there are several ways of encoding a possessive relation between two entities, including (1) predicative possessive constructions, (2) external possessive constructions, and (3) adnominal possessive constructions. Of these three types of constructions, only adnominal possession is modeled by this library, but I define each of them briefly here in order to more clearly define the scope of this thesis.

**Predicative possessive expressions** are ones in which the possessive relationship between two entities is expressed by means of a verb. This is often a verb that is roughly equivalent to the English verbs *have* or *belong*. Alternatively, in some cases, possession may be expressed by constructions where the possessor is marked by an oblique case or an adposition and the verb is an existential verb of some kind, as in the Latin example in (1) below.

- (1) mihi        est                liber  
       1SG.DAT be.3SG.PRES book.NOM  
       ‘I have a book’ (lit. ‘to me is a book’) [lat] (Heine, 1997, 32)

**External possessive constructions** are defined as expressions in which “a possessive modifier does not occur as a dependent constituent of the modified NP, but NP-externally as a constituent of the clause” (Haspelmath, 1999, 109). This can be seen in examples such as (2), from German. Here, the possessor (*dem Kind* ‘the child’) is not a dependent of the possessum, but rather is simply another argument of the verb (*wusch* ‘washed’). However, in this construction, it is only understandable as the possessor. External possessive expressions are not found in all languages and they are not modeled by this library.

- (2) Die Mutter wusch dem Kind die Haare.  
 the mother washed the.DAT child the.ACC hairs  
 ‘The mother washed the child’s hair.’ [ger] (Haspelmath, 1999, 109)

In **adnominal possessive constructions**, the possessor and the possessum form a single noun phrase together (Koptjevskaja-Tamm, 2002, 141). The possessor is a syntactic dependent of the possessum, though it may not always be of the same class of dependent, as discussed in chapter 3 below. An example of an adnominal possessive phrase in Finnish is given in (1) below.

- (3) heidän ystävä-nsä  
 3PL.POSS friend-3POSS  
 ‘their friend’ [fin] (Toivonen, 2000, 585)

The library described in this thesis only covers adnominal possessive phrases. Having described the scope of the present work, I turn in the next chapter to some background on the Grammar Matrix and the formalisms that it employs.

## Chapter 2

### BACKGROUND

The present chapter gives some necessary background on the Grammar Matrix and the formalisms that underpin it. The topics covered here include the syntactic theory that undergirds the Grammar Matrix, Head-driven Phrase Structure Grammar (section 2.1); a formalism for encoding linguistic semantics, Minimal Recursion Semantics (section 2.2); and the functioning of the Grammar Matrix itself (section 2.3).

#### **2.1 Head-driven Phrase Structure Grammar**

Head-driven Phrase Structure Grammar (HPSG) is a grammatical system that was developed by Pollard and Sag (1994) out of Generalized Phrase Structure Grammar (Gazdar, 1985). HPSG is a highly lexicalized grammatical system, with an emphasis on modeling surface forms without the use of concepts such as transformations and movements. The bulk of the description that follows pertains to HPSG broadly, though some of the features mentioned are specific to the variant of HPSG employed in the Grammar Matrix.

Entities such as words and phrases in HPSG are modeled as *feature structures*, or collections of features paired with their values. A given feature may take a single atomic value or its value may be a bundle of other features. For example, in a feature structure representing a noun, the feature NUM might appear, with the atomic value *sg*. At the same time, the feature VAL (representing the valence properties of a word or phrase), doesn't take a single atomic value, but instead takes a collection of features as its value, including COMPS (representing the entity's complements), SPR (representing specifiers), and SUBJ (representing subjects). These nested feature structures are typically represented as an attribute value matrix (or AVM), as shown in (4) below. An explanation of the features shown will follow.

The feature structures in HPSG are *typed* feature structures. That is, a given feature structure belongs to a type (which is indicated visually by having the name of the type appear in the top left of the attribute-value matrix), which has certain constraints on it. For example, the AVM in (4) is of type *word*. The type *word* carries the constraint that it must have a HEAD value of some kind. These types are arranged in an inheritance hierarchy, where the constraints on a parent are inherited by its children. So, any child of the type *word* would also be constrained to have a HEAD value.

$$(4) \left[ \begin{array}{c} \textit{word} \\ \text{SYNSEM} \left[ \begin{array}{c} \text{LOCAL} \left[ \begin{array}{c} \text{CAT} \left[ \begin{array}{c} \text{HEAD} \left[ \begin{array}{c} \textit{noun} \\ \text{MOD} \langle \rangle \end{array} \right] \\ \text{SUBJ} \langle \rangle \\ \text{SPR} \langle \rangle \\ \text{SPEC} \langle \rangle \\ \text{COMPS} \langle \rangle \end{array} \right] \\ \text{VAL} \left[ \begin{array}{c} \text{CONT} \left[ \begin{array}{c} \text{HOOK} \left[ \begin{array}{c} \text{INDEX} \left[ \begin{array}{c} \text{PNG} \left[ \begin{array}{c} \text{NUM } \textit{sg} \\ \text{PER } \textit{3rd} \end{array} \right] \end{array} \right]$$

An attribute-value matrix like the one in (4) can represent a word, a lexical rule, or a phrase structure rule. In more complex AVMs, many other features appear, but the features shown above are particularly important for the analysis that will be presented in chapter 4. The feature *CAT* contains the syntactic information about this word, while the feature *CONT* contains the semantic information. Of particular note within *CONT* are the features *INDEX* and *PNG*. *INDEX* takes as its value a variable that points to the semantic contribution of the sign. This helps model semantic composition by exposing a variable representative of the whole of the sign's semantics. The feature *PNG* contains elaborated semantic information — in this case, the information that this word is singular and third person. Within the *CAT* features, of note are the *MOD* and *VAL* features. The *VAL* features encode the arguments this constituent can take in the form of lists. All these lists are shown as empty in (4) above, which indicates that the word is not looking for any arguments. Were this word in

need of a specifier, say, then an AVM representing this word’s specifier would appear on its SPR list. Each of the features under VAL encodes a slightly different kind of argument: SPR holds a list of needed specifiers; COMPS, a list of needed complements; and SUBJ, a list of subjects.<sup>1</sup> Since a given constituent can have at most one specifier or subject, the SPR and SUBJ lists have a maximum length of one. A feature SPEC is also given, which is the converse of the SPR feature—when it is non-empty, it indicates that the word may serve as the specifier for a constituent that matches the specifications of the item on the SPEC list. The addition of the SPEC list allows a specifier to constrain the types of things it can specify. The SPEC list also has a maximum length of one. Lastly, the feature MOD figures prominently in the present analysis. This feature is also list-valued, also of maximum length of one, and when non-empty, indicates that the word may modify a constituent that matches the feature constraints of the item on the MOD list.

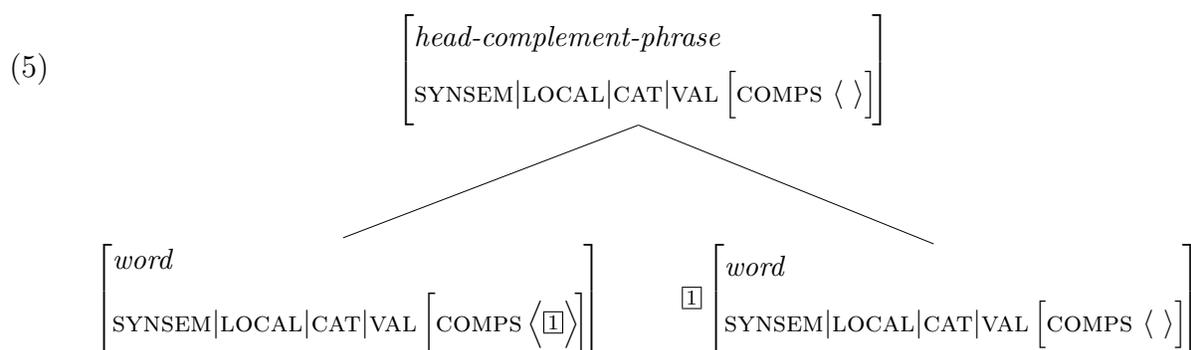
For each of these list-valued syntactic features, there are one or more corresponding phrase structure rules which build phrases out of constituents with a non-empty list value for the appropriate feature. For example, the COMPS list corresponds to a head-complement rule,<sup>2</sup> which is shown licensing a phrase in (5) below. The head-complement rule takes two constituents, one of which has a non-empty COMPS value (called the head daughter of the phrase), and the other of which has constraints which match the first element on the non-empty COMPS list (called the non-head daughter). The phrase structure rule then licenses the combination of these two constituents into a phrase, which is represented as an AVM with the same VAL features as the head daughter, save for the COMPS list, which has had one element removed. Put another way, the head-complement rule is the mechanism by

---

<sup>1</sup>Since subjects and specifiers appear in what superficially seem to be the same distributional contexts, it is important to distinguish the two. Pollard and Sag (1994, 359) state that the distinction consists of the fact that specifiers “lack the potential to be semantic arguments; instead their semantic contribution is more abstract, typically quantificational or degree-denoting in nature [...] Another difference, evidently related, is that the specifier positions do not seem to be available for obligatory control as subjects are; nor do we find raising of specifiers to an argument position of a higher predicate.”

<sup>2</sup>Throughout this thesis, phrase structure rules are italicized when they refer to rules as implemented in the Grammar Matrix, and given in plain type when they refer simply to a general rule or category of rules, not their implemented form.

which a constituent, the head daughter, selects for and combines with its complement, the non-head daughter. In order to indicate that a value is shared between two features in an attribute-value matrix, their values are labeled with a shared numerical tag, e.g.  $\boxed{1}$ . Note that, as a headed phrase structure rule, the head-complement rule also has the property that its head daughter shares its HEAD value with the head value of its parent, though this is not shown explicitly below.



Other MOD and VAL features correspond to phrase structure rules that function similarly: the head-subject rule combines a constituent that has a non-empty SUBJ value with a constituent that is compatible with that SUBJ value. The head-modifier rule works in largely the same fashion, with the notable difference that although the modifier selects for the thing it modifies via its MOD list, it is the non-head daughter. As such, it does not pass its VAL or HEAD features up to the phrase level. Lastly, the head-specifier rule functions the same as the head-complement or head-subject rules, but where in those rules, just one constituent selected for the other, in a head-specifier phrase, the head selects for its specifier via the SPR list, while the specifier selects for its head via the SPEC rule. Note that in all cases, the head daughter of any of these headed phrase rules shares its HEAD features with the parent of the rule.

In order for these phrase rules to apply to any pair of constituents, the relevant constraints must *unify*. For example, in any case where the phrase rule identifies feature values together, as with the tag  $\boxed{1}$  in (5), the feature values of these constituents must be either 1) identical, or 2) underconstrained by at least one constituent. While the first of these cases is intuitive enough, the second deserves some clarification. If one constituent requires, for example, that

its complement have the feature value [ CASE *nom* ], its complement could be a noun that simply is not specified to have any particular CASE value. An underconstrained CASE value is compatible with any value for CASE and will therefore unify. In addition to those features that are identified via tags, each of the signs that serve as the daughters of the rule has to unify with the constraints imposed by the rule on that daughter. Each daughter of the rule may also place constraints on its sister.

In HPSG, individual lexical items and rules are defined not as single, unrelated entities, but as entities that inherit their constraints from other entities. For example, every noun in a grammar is not defined separately and fully as a distinct type. Rather, there is a supertype, called *noun-lex*, which carries all the constraints common to all nouns. Different subtypes of nouns inherit all the constraints that *noun-lex* has and then add constraints of their own—for example, in English, common nouns inherit from *noun-lex*, but must then add the constraint that they are all third person. Any given type unifies with its supertypes; two types may also unify in the case where they share a subtype, even if neither is a supertype of the other.

In the particular variant of HPSG that the Grammar Matrix implements (known as the DELPH-IN<sup>3</sup> joint reference formalism), any given grammar entity (or *instance*) can only instantiate one type. However, any given type may itself have multiple supertypes. This system of multiple inheritance not only makes for a much more efficiently written and readable grammar, but it also captures a linguistic generalization—words and phrases can be grouped into cross-cutting classes that share certain behaviors.

These mechanisms underpin all Grammar Matrix grammars, which are essentially collections of typed feature structures representing lexical items, lexical rules, and phrase rules, related to each other through inheritance hierarchies as described here. These grammars can be used to parse strings and, with the addition of the semantic framework laid out in the

---

<sup>3</sup>DEep Linguistic Processing with HPSG - INitiative (DELPH-IN) is a community of researchers whose work concerns computationally focused HPSG research, making use of shared conventions, formalisms and software (Oepan et al., 2002).

following section, map these strings to a representation of their meaning. Grammars can also be used in the opposite direction, to generate strings from semantic representations

## 2.2 *Minimal Recursion Semantics*

In order to model not only the syntactic structure of a language but the corresponding semantic structures, a suitable formalism for representing meaning must be selected. The Grammar Matrix makes use of Minimal Recursion Semantics (MRS), a system developed for use in broad-coverage feature structure grammars (Copestake et al., 2005). Each valid syntactic parse maps to one or more semantic representations in the MRS formalism. The objective of the developers of MRS was to create a system that encoded roughly the same information that predicate logic may encode, but in a fashion that was readily compatible with feature structure grammars (Copestake et al., 2005, 291).

The MRS formalism was also designed with the goal of dealing with the issue of quantifier scope ambiguity. This ambiguity arises with constructions with multiple quantifiers, as in classic examples such as *Every dog chases some white cat* (given in Copestake et al. 2005), where a reading with wide-scope *every* is compatible with a scenario in which a cat exists for each dog, and a reading with wide-scope *some* is compatible with a scenario in which one cat exists that is chased by all dogs. This ambiguity will occur in all cases where quantifiers exist within the possible scope of one another. In a system like predicate logic, both of these possible readings would have to be given separately, meaning in many cases, a single valid syntactic parse would map to many possible semantic representations. MRS aims to solve this problem by encoding this scope ambiguity in a correspondingly ambiguous manner, and thus allowing a one-to-one mapping of syntactic parses to semantic representations in many cases where a one-to-many mapping would otherwise be required.

A primary part of the architecture of an MRS representation is an unordered collection of *elementary predications* or *relations*, which are found in a list called RELS. Unlike in predicate logic, these relations are never embedded inside one another. Instead, each predication contains not only its own arguments, but also a LBL value, which serves as a handle for that

predication. An example of an elementary predication for a noun is given in (6) below.

$$(6) \quad \begin{bmatrix} \text{\_cat\_n\_rel} \\ \text{LBL} & h0 \\ \text{ARG0} & x1 \end{bmatrix}$$

This LBL value can be used by other predications to describe scopal relations. Elementary predications that share the same scope have the same label; elementary predications that take scopal arguments use labels as the values of those arguments. For example, instead of a noun’s primary elementary predication being embedded inside its quantifier’s predication, the quantifier simply refers to the LBL of the noun’s predication.

However, in order to preserve the ambiguous representation of scope discussed above, in many cases the handle of one predication—say, a noun’s primary predication—is not directly used as the argument of another predication, such as the quantifier predication. Instead, another structure is introduced: a list of *handle constraints*, abbreviated HCONS. This list consists of *equal modulo quantifiers* relations (abbreviated *qeq*). These relations are used to communicate that, rather than one predication being equal to the thing inside another predication, it is equal modulo quantifiers. For example, if a quantifier predication takes another predication’s LBL value as an argument, other quantifiers may have scope that places them between the quantifier and the predication it quantifies. This means that we can encode the fact that *not* negates *see* in the example sentence *Kim did not see every cat* by having the ARG1 value of the *neg\_rel* be *qeq* to the ARG0 value of *see\_rel*. Since the two values are *equal modulo quantifiers*, the quantifier *every* may have scope between the two, or it may not, generating the two readings (one where there are some cats Kim does not see, and one where there are no cats that Kim sees). In this way, by modeling the ambiguity present in the language itself, MRS avoids the explosive one-to-many mapping between a single syntactic parse and the potentially many representations of that parse which vary only in scope.

Example (7) below shows a RELS list and an HCONS list being used to model quantification in the noun phrase *the cat*. In this example, the RSTR value (which stands for *restriction*) of

the quantifier relation functions as one argument of the *qeq* on the HCONS list, while the LBL value of the noun relation functions as the other. This can be interpreted as meaning that, barring any other quantifiers, the *cat* relation is equal to the RSTR value of the quantifier.

$$(7) \left[ \begin{array}{l} \text{RELS} \left\langle \begin{array}{l} \textit{exist\_q\_rel} \\ \text{LBL} \quad h1 \\ \text{ARG0} \quad x0 \\ \text{RSTR} \quad h2 \\ \text{BODY} \quad h3 \end{array} \right\rangle, \left\langle \begin{array}{l} \textit{\_cat\_n\_rel} \\ \text{LBL} \quad h0 \\ \text{ARG0} \quad x0 \end{array} \right\rangle \\ \text{HCONS} \left\langle \begin{array}{l} \textit{qeq} \\ \text{HARG} \quad h2 \\ \text{LARG} \quad h0 \end{array} \right\rangle \end{array} \right]$$

In addition to the RELS and HCONS lists, an MRS has an LTOP value and an INDEX value. The LTOP value functions essentially as a handle to the whole MRS for purposes of modeling scope. The INDEX and LTOP are of particular importance for modeling semantic composition, since they represent the whole of the semantic information encoded in the MRS representation and can be ‘exposed’ to the rest of the grammar in order to model semantic composition. A full MRS for the noun phrase *the cat* is shown in (8) below:

$$(8) \left[ \begin{array}{l} \text{LTOP} \quad h4 \\ \text{INDEX} \quad x0 \\ \text{RELS} \left\langle \begin{array}{l} \textit{exist\_q\_rel} \\ \text{LBL} \quad h1 \\ \text{ARG0} \quad x0 \\ \text{RSTR} \quad h2 \\ \text{BODY} \quad h3 \end{array} \right\rangle, \left\langle \begin{array}{l} \textit{\_cat\_n\_rel} \\ \text{LBL} \quad h0 \\ \text{ARG0} \quad x0 \end{array} \right\rangle \\ \text{HCONS} \left\langle \begin{array}{l} \textit{qeq} \\ \text{HARG} \quad h2 \\ \text{LARG} \quad h0 \end{array} \right\rangle \end{array} \right]$$

In Grammar Matrix grammars, the components of the MRS are incorporated into the feature structures that are already built to model lexical items and rules of various kinds.

Elementary predications can be introduced by any of these elements in the grammar. The MRS representation of a given expression is built compositionally from the MRSs of its constituents, meaning that all the elements of the constituents' MRSs must be present in the final expression's MRS. The exact way in which these predications interact is determined by constraints encoded in individual rules and lexical entries.

When combined with the HPSG feature-structure grammars described in section 2.1 above, the MRS formalism allows for grammars to model not only syntactic, but also semantic structure. In the next section, I discuss how the LinGO Grammar Matrix aids in producing these MRS-augmented HPSG grammars.

### **2.3 The LinGO Grammar Matrix**

Hand-engineered grammars represent one of the most information-rich tools available in natural language processing and linguistic research. However, it can take years to build a grammar with a useful level of coverage. The LinGO Grammar Matrix exists in order to expedite this grammar creation process, making implemented grammars a more practically accessible resource (Bender et al., 2002).

The Grammar Matrix consists of a core set of grammatical type definitions and a customization system. The core definitions in the Grammar Matrix are designed to cover a maximally broad set of linguistic phenomena, and so function not only as a practical aid in grammar engineering, but also as a set of claims about what typological generalizations can be made about various phenomena. This set of grammatical type definitions can be combined and augmented by the customization system to model a wide range of phenomena (Drellishak, 2009; Bender et al., 2010). This customization system consists of a web questionnaire, which asks the user-linguist to input information about the language to be modeled, such as basic word order, case behavior, and so forth. The answers to this questionnaire are stored in a text file, referred to as a *choices file*. This choices file is then parsed by back-end Python code, which creates a grammar according to the user's specifications.

The main components of the grammar produced by the customization system are as

follows: a file containing the core matrix type definitions (`matrix.tdl`), a file containing types that inherit from and extend the matrix types (which is given the name of the language being modeled), and then finally files instantiating lexical items (`lexicon.tdl`), spelling-changing rules (`irules.tdl`), non-inflecting lexical rules (`lrules.tdl`), and phrase structure rules (`rules.tdl`). These are all written using the DELPH-IN joint reference formalism (sometimes also called Type Description Language (TDL)), whose conventions are described in Copestake (2002), which provide a standardized way of writing feature-structure grammars that is interpretable by all DELPH-IN processors.

The customization system has been expanded over the years by the addition of libraries, each of which covers a particular phenomenon. Previous libraries have covered coordination (Drellishak and Bender, 2005), morphotactics (O’Hara, 2008; Goodman, 2013), argument optionality (Saleem, 2010), tense and aspect (Poulson, 2011), information structure (Song, 2014), negation (Crowgey, 2012), adjectives (Trimble, 2014), evidentiality (Haeger, 2017), and valence-changing operations (Curtis, 2018). The adnominal possession library interacts significantly with the morphotactics library (O’Hara, 2008), since many languages mark possessive phrases with inflectional morphology. For this reason, it is necessary to give some background on this library in particular in order to fully understand the functioning of the present library.

The goal of the morphotactics library is to allow the user to model concatenative morphological processes, including such things as which morphemes cannot, must, or may cooccur, and the relative ordering of morphemes (O’Hara, 2008, 24). To this end, the user can define any number of *position classes* for nouns or verbs. These position classes correspond to a slot in traditional templatic morphology — a set of affixes that occur in the same position relative to other morphemes but which cannot cooccur. The user can specify that any given position class attach only to a certain class of words (say, a user-defined class of pronouns), or only attach after another affix has applied (say, a number-marking suffix). For each position class, the user may define any number of *lexical rule types*, which correspond to affixes (or classes of affixes) that occur in that position (including zero-inflecting affixes). These lexical

rule types may be specified to carry certain features, such as singular number, nominative case, and so forth.

A position class may be specified to be either required or optional, meaning that any word to which this position class applies must have acquired one of the affixes at this position class before it is permitted to participate in phrase structure rules. The morphotactics library models the obligatoriness of certain position classes by defining FLAG features located at the INFLECTED feature. Each position class corresponds to one FLAG feature, which is set to have a negative value in all uninflected lexemes. Passing through any instance of a lexical rule type defined at that position class sets this FLAG value to positive. All phrase structure rules require that any FLAG feature corresponding to an obligatory position class have the value *na-or-+* — that is, it must be either set to positive (having passed through an appropriate lexical rule) or the lexeme must be of a type to which the position class does not apply.

Making use of these functions within the morphotactics library, the adnominal possession library can model the behavior of possessive affixes, as well as modeling their interaction with other affixes. This means that defining the morphotactic behavior of possessive affixes occurs on a different subpage of the customization questionnaire, and is partially handled by code that belongs to another library, though with necessary adaptation made, as detailed in chapter 5.

## **2.4 Summary**

In this chapter, I have discussed the theoretical underpinnings of the Grammar Matrix, including Head-Driven Phrase Structure Grammar and Minimal Recursion Semantics. I have also described the functioning of the Grammar Matrix and the customization system. In the coming chapter, I describe the various types of adnominal possessive constructions that are attested in the world's languages, so as to better define the typological space under discussion.

## Chapter 3

## TYPOLOGY OF ADNOMINAL POSSESSION

In this chapter, I review the variety of adnominal possessive constructions attested in the world's languages. The information in this chapter is based largely on Grashchenkov (2005), Dryer and Haspelmath (2013), and Koptjevskaja-Tamm (2002). First, in section 3.1, I discuss some common terminology that I use throughout to describe adnominal possessive constructions. Then in section 3.2, I discuss the range of meanings that can be conveyed by adnominal possessive constructions. Lastly, in section 3.3, I describe the range of syntactic structures used to encode possessive constructions.

**3.1 Terminology**

At their most basic, possessive constructions denote some binary relationship between two entities. The nouns or pronouns which refer to these entities are generally termed the *possessor* and the *possessum*. The *possessum* denotes the entity that is the property of, at the disposal of, or part of the other entity (Koptjevskaja-Tamm, 2002, 141). It is also sometimes called the *possessee* or the *head* of the possessive phrase. In (9) below, the possessum is shown in bold:

(9) the cat's **paw** [eng]

Welsh:

**car** y meddyg

**car** the doctor

'the doctor's **car**' [wel] (Koptjevskaja-Tamm, 2002, 144)

Archi:

šopil-li-n                      **mašina**

driver-OBL.SG-GEN **car**

'the driver's **car**' [aqc] (Koptjevskaja-Tamm, 2002, 143)

The *possessor*, also called the *dependent* of the possessum, is the noun or pronoun denoting the entity which owns, controls, or subsumes the other entity (Koptjevskaja-Tamm, 2002, 141). In (10) below, the possessor is shown in bold:

- (10) English:  
**the cat's** paw [eng]  
 Welsh:  
 car **y meddyg**  
 car **the doctor**  
 ‘**the doctor's** car’ [wel] (Koptjevskaja-Tamm, 2002, 143)  
 Archi:  
**šopil-li-n** mašina  
**driver-OBL.SG-GEN** car  
 ‘**the driver's** car’ [aqc] (Koptjevskaja-Tamm, 2002, 143)

### 3.2 *Semantics of possession*

A wide range of meanings can be conveyed by what are often called possessive noun phrases. In this section, I outline the most common meanings that are associated with possessive phrases. In general, in this thesis I use the terms *possessor* and *possessum* to refer the two major **syntactic** roles in the possessive phrase, rather than to any particular semantic relationship of the possible ones outlined below. However, since the semantic status of constituents is at issue here, in the present section I use the more semantically neutral terms *head* and *dependent* to refer to the possessum and the possessor, respectively. All examples given here are in English, but these phenomena are attested in many languages (Koptjevskaja-Tamm, 2002).<sup>1</sup>

---

<sup>1</sup>In some languages there may be constructions that are syntactically identical to possessive phrases, but which have entirely distinct semantic representations. These include constructions where the head is an argument-taking noun, such as *photograph*. In this case, the dependent *Kim* in the phrase *Kim's photograph* is understood as the person depicted in the photograph, rather than its owner. Also in this category are deverbal nouns such as *demonstration*, which assigns thematic roles to its dependents. Thus in the phrase *my demonstration of the procedure*, the first person argument is understood as the agent of the act of demonstrating. Though these constructions are syntactically identical to possessive constructions, they are sufficiently distinct semantically as to require an entirely different semantic representation, and so are considered outside the scope of this present work.

### 3.2.1 *Range of attested meanings*

**Ownership** One possible relationship between dependent and head is one of ownership, where the head, which is prototypically inanimate, is the literal property of the prototypically animate dependent:

- (11)       the teacher's house  
               Kim's glasses  
               your PC

**Part-to-whole relations** Possessive or possessive-like constructions can also convey a partitive relationship, where the head refers to some portion of a whole denoted by the dependent (Koptjevskaja-Tamm, 2003, 537):

- (12)       the rest of the ice cream  
               the play's first act

The relationship between a person (or other animate being) and their body parts is a particularly salient subtype of the part-to-whole relation (Koptjevskaja-Tamm, 2002, 141).

- (13)       the cat's paw  
               Kim's elbow  
               my hair

**Kinship relations** Possessive or possessive-like constructions can also convey symmetrical and asymmetrical kinship relations between the entities denoted by dependent and head (Koptjevskaja-Tamm, 2002, 141):

- (14) Symmetrical:  
               my sibling  
               Pat's cousin

- (15) Asymmetrical:

her grandmother  
 Kim's niece

**Modification by non-referential dependent** The dependent may act non-referentially in some cases, as what Peters and Westerstahl term a modifier, than as a literal owner of the head (Peters and Westerstahl, 2013, 715):

- (16)      women's clothing  
             a farmer's tan

### 3.2.2 *Defining the scope of meanings to be covered by the library*

For the purposes of implementing a library for adnominal possession in the Grammar Matrix, only those possessive phrases that can be adequately modeled by including a relation with the *poss\_rel* relation will be considered in-scope. The *poss\_rel* relation takes the following form, where ARG1 is identified with the possessum and ARG2 with the possessor:

- (17)      
$$\left[ \begin{array}{ll} \text{PRED} & \textit{poss\_rel} \\ \text{LBL} & h \\ \text{ARG0} & e \\ \text{ARG1} & x_1 \\ \text{ARG2} & x_2 \end{array} \right]$$

The *poss\_rel* corresponds to a relatively underspecified relation of possession between the ARG2 (possessor) and ARG1 (possessum). As such, it can adequately model cases such as ownership, body part possession, kinship relations, and cases where the possessor is a non-referential modifier of the possessum.

### 3.3 *Syntactic exponence of possession*

In this section, I describe the various syntactic constructions used to encode adnominal possessives in the world's languages. The material here is based largely on the work of Grashchenkov (2005), Dryer and Haspelmath (2013), and Koptjevskaja-Tamm (2002). I

cover constructions both with and without overt morphology marking them as possessive phrases, covering what form these markings take, where they appear, and any other grammatical features they may carry. I also discuss other important typological distinctions in types of possessive phrases, such as the distinction between determiner-like and modifier-like possessors, and between inalienably and alienably possessed nouns.

Note that any given language may employ only one type of possessive construction or may allow several different types of possessive constructions. Sometimes different possessive constructions are used in a single language to express slightly different meanings (see section 3.2 above), and in other cases, strategies are restricted in their use by syntactic properties of the possessor and the possessum (e.g. membership in a certain lexical class). In this section, I am primarily concerned with describing the range of possible constructions, rather than the combinations of constructions that appear in any given language.

### 3.3.1 *Unmarked constructions*

The simplest possessive constructions are those wherein there is no overt morphological marking of the phrase as possessive. In these constructions, the possessor and possessum are simply juxtaposed, with their respective roles being determined by their relative ordering, as in the following examples from Yoruba:

- (18) íwè baba  
 book father  
 ‘father’s book’ [yor] (Grashchenkov, 2005, 28)

- (19) iran awon aomo oduduwa  
 descendant PL child Oduduwa  
 ‘descendants of the children of Oduduwa’ [yor] (Grashchenkov, 2005, 28)

A given language may or may not impose feature constraints on the possessor in an unmarked construction. For example, the possessor may be required to appear in a certain case. Since this marking is not unique to possessive constructions, it is still analyzed as

an unmarked possessive construction. See section 4.2.1 below for further discussion on this point.

### *3.3.2 Marked constructions*

In those constructions that do contain overt morphological marking that designates them as possessive, the marking may appear in a number of different locations and take the form of an affix, clitic, or adposition. In this section, I discuss the location of possessive markings, and the other features, such as case and agreement, that these possessive markings may carry.

#### *3.3.2.1 Location of possessive marking*

**Possessor marking** Marking the possessor in some manner is the “most typologically widespread means of encoding possession” (Grashchenkov, 2005, 31). Many languages, including English, employ constructions where the possessor is marked by some affix, clitic or adposition:

Adpositions:

(20) Bulgarian:

lah na proletta  
breath POSS spring

‘the breath of spring’ [bul] (Grashchenkov, 2005, 31)

(21) Danish:

smag-en af maelk  
taste-DEF POSS milk

‘the taste of milk’ [dan] <sup>2</sup>(Grashchenkov, 2005, 31)

Clitics:<sup>3</sup>

(22) Basque:

neska gazte-a=ren edertasuna  
 girl young-DEF=POSS beauty  
 ‘the beauty of the young girl’ [baq] (Grashchenkov, 2005, 33)

(23) English:

John and Mary=’s child [eng] (Grashchenkov, 2005, 33)

Affixes:

(24) Malagasy:

zana d-rabe  
 child POSS-Rabe  
 ‘the child of Rabe’ [mlg]

(25) Imbabura Quechua:

José-paj wasi  
 José-POSS house  
 ‘José’s house’ [qvi] (Grashchenkov, 2005, 34-35)

In possessor-marking constructions, the possessor may in some cases be a personal pronoun taking a special possessive form. This pattern is common in Indo-European and other languages:

(26) a. English:

her speech

b. Kiribati:

ara auti  
 1PL.POSS house

---

<sup>2</sup>The possessive markers *na* in Bulgarian and *af* in Danish are analyzed as by Grashchenkov (2005) as possessor-marking rather than possessum-marking. This is presumably because they are analyzed as adpositions, and both Bulgarian and Danish have prepositions and no postpositions (Morse, 1859; Lundskaer-Nielsen and Holmes, 2011), and there are no other considerations to prompt an alternative analysis in these cases.

<sup>3</sup>In this thesis, the term ‘clitic’ is used to refer to a unit that is fully syntactically independent, but phonologically dependent on another word.

‘our house’ [gil] (Dryer, 2013, 235)

**Possessum marking** Marking of the possessum is also “extremely widespread typologically” (Grashchenkov, 2005, 35). However, according to Grashchenkov (2005), there are no cases where the possessum is marked by a unique possessive marker which carries no additional information. Instead, there are two ways in which the possessum may be marked. First, the possessum can be marked as agreeing with the possessor in features such as person and number (see section 3.3.2.2 below). Second, the marking that appears on the possessum may be a generic marking that appears on nouns that have dependents of various kinds, not just possessors. The following examples from Farsi demonstrate how the marking on the possessum coincides with the marking on a noun being modified by an adjective:

(27) mej-e sorx  
 wine-EZ red<sup>4</sup>  
 ‘red wine’

(28) mej-e daneSDu  
 wine-EZ student  
 ‘the student’s wine’ [per] (Grashchenkov, 2005, 36)

Following are some examples from Yucatec Maya and Aalsea of cases where the possessum is marked as agreeing with the possessor, a phenomenon discussed at greater length in section 3.3.2.2 below. In both languages, the possessum agrees in person with the possessor:

(29) Yucatec Maya (clitic marker):  
 u=k‘àaba‘ le x-ch‘up-pàal-a‘  
 POSS.3=name DEF FEM-woman-child-D1  
 ‘the name of that girl’ [yua]

(30) Aalsea (affix marker):  
 a-s xámni c-laqúsin-ķ  
 this-DEF whale POSS.3-skin-POSS.3

---

<sup>4</sup>The gloss EZ here is in reference to the *ezāfe*, the traditional term in Farsi linguistic literature for this morpheme, which functions as a ligature between two words.

‘the skin of this whale’ [aes] (Grashchenkov, 2005, 36-37)

Éwé has constructions that are analyzed by Grashchenkov (2005) as possessum-marking, with the marker taking the form of an adposition. However, in this case, the possessum marker coincides with the possessive pronoun and agrees with the possessor:

(31) Éwé:

lã wó afə  
beast his(3.SG.POSS) paw

‘the paw of the beast’ [ewe] (Grashchenkov, 2005, 36)

In some possessum-marking constructions where the possessor is understood to be a dropped pronoun, no possessor need appear at all. This can be seen in the following examples from Hungarian, where the overt pronoun *mi* ‘we’ can be dropped:

(32) Hungarian:

a. a mi kalap-unk  
the we-NOM hat-POSS.1PL  
‘our hats’ [hun] (Szabolcsi, 1994, 8)

b. a kalap-unk  
the hat-POSS.1PL  
‘our hats’ [hun] (Szabolcsi, 1994, 8)

**Double-marking** There are many languages that have possessive constructions where both the possessor and the possessum are marked in some fashion. In many cases, one or more of these markings also carries agreement features that agree with the other entity in the phrase (see section 3.3.2.2 for more on agreement within the possessive phrase). In the Adyghe example shown in (33) below, the ergative case suffix serves as a possessor-marker,<sup>5</sup> while the possessum is marked with a third person agreement marker. Similarly, in the Komi example shown in (34), the word *rajon* ‘area’ carries a suffix marking it as possessor, while the possessum optionally carries a suffix indicating that it agrees with the possessor in person and number.

(33) Adyghe:

č'elejeraže-m ə-s<sub>w</sub>əz  
 teacher-ERG 3.POSS-wife  
 'the teacher's wife' [ady]

(34) Komi:

rajon-len glava(-ys)  
 area-POSS head(-3SG.POSS)  
 'head of the area' [kom] (Grashchenkov, 2005, 38)

Finnish provides examples of constructions with double-marking in the case where the possessor is a pronoun. These pronominal possessors are marked as being possessors, but an agreeing possessum marking is required as well.

(35) Finnish:

- a. minun velje-ni  
 my brother-POSS.1SG  
 'my brother' [fin] (Karlsson, 2015, 125)
- b. velje-ni  
 brother-POSS.1SG  
 'my brother' [fin] (Karlsson, 2015, 125)

**Second-position marking** Though it is significantly less common than the marking patterns described above, possession marking in the second position is attested in several of the world's languages. In these constructions, the morpheme that marks the phrase as possessive appears after the first element of the noun phrase, whatever that element may be. Because this particular construction is quite rare and also very different from other possessive constructions, it is not covered by the analysis and implementation detailed in this thesis. For example, in Tagalog, a possessor pronoun may appear to the right of the head (e.g. the first person singular pronoun clitic =*ko* in (36a)), or it may appear in second position (e.g. in (36b), where it cliticizes not to the head noun, but to the negation word *hindi*):

(36) Tagalog

- a. Mahflig=ako sa=maɲa=bátà basta hindi anak=ko  
 like=1SG.NOM OBL=PL=KID just NEG child=1SG.GEN  
 ‘I like kids, so long as they’re not kids of mine’ [tgl] (Kaufman, 2010, 221)
- b. hindi=ko=iyón problema  
 NEG=1SG.POSS=that.ABS problem  
 ‘That’s not my problem’ [tgl] (Aldridge, 2004, 262)

### 3.3.2.2 Other features carried by possessive markers

**Agreement of possessum with possessor** As noted in section 3.3.2.1 above, many languages that mark the possessum require (or in some cases, optionally allow) it to agree with the possessor in person, number or gender. The following examples illustrate the agreement of possessum with possessor in person and, in the case of Hungarian, number:

- (37) Hungarian:  
 Mari-nak a kalap-ja(-i)  
 Mari-DAT the hat-POSS-PL(-3SG)  
 ‘Mari’s hats’ [hun] (Szabolcsi, 1994, 2)

- (38) Finnish:  
 heidän ystävä-nsä  
 their friend-3POSS  
 ‘their friend’ [fin] (Toivonen, 2000, 585)

**Agreement of possessor with possessum** Constructions where the possessor agrees with the possessum in person, number and/or gender features are also well-attested. The following example from Romani shows gender agreement between possessor and possessum:

- (39) Romani:  
 e manús-es-quiri buzni  
 the:OBL.M.SG man-OBL.SG.M-GEN.:F.SG.NOM goat(FEM)  
 ‘the man’s goat’ [rom] (Koptjevskaja-Tamm, 2001, 962)

In some languages, particularly languages of Australia, it is also not uncommon for the possessor to agree with the possessum in case, even if the possessor is already assigned an NP-internal case, such as the genitive. Gumbaynggirr and Old Georgian both show this pattern, with the possessor carrying both a genitive case suffix assigned from within the noun phrase and a case agreement suffix:

- (40) Gumbaynggirr:  
 ba:ba-gu gunuy-gundi-yu ju:ngu jala:ny jamay barway  
 father-ERG child-GEN-ERG say.FUT tongue.NOM too big.NOM  
 ‘The child’s father will say, “(Your) mouth is too big.”’ [kgs] (Malouf, 2000, 204)

This same agreement pattern is found in Old Georgian:

- (41) Old Georgian:  
 cq’oba-sa mt’er-ta-sa  
 attack-DAT enemy-GEN.PL-DAT  
 ‘to the enemies’ attack’ [oge] (Grashchenkov, 2005, 39)

This pattern of case agreement leading to double case marking is not modeled in the library.

**Relationship of possessive marking to case system** In many linguistic traditions, the marking that is borne by a possessor is analyzed as part of the case system, often as a case called the genitive which is assigned internally to the noun phrase. In some languages, the possessor marking is coincident with a different core case, such as the dative in Hungarian (see (37) above), but while the name of the case may differ, the possessor marking is still analyzed as part of the case system in these languages.

However, possessum markers often cannot be analyzed as belonging to the case system. As noted in section 3.3.2.1 above, in all languages studied by Grashchenkov (2005), any affix whose sole function was to mark the possessum was found to carry agreement information (Grashchenkov, 2005, 35). For example, in the Hungarian sentence shown in (42), the possessum is marked as having a 1SG possessor. A case affix cannot also carry agreement information, and so the possessum-marking affix must be analyzed as distinct from the case system in these languages.

- (42) kivasalom      az ing-em-et  
 iron.NPST.1SG the shirt-1SG.POSS-ACC  
 ‘I (will) iron my shirt’ [hun] (Rounds, 2001, 293)

Various analyses are possible for the scenario where a case marker coincides with a possessor marker (see section 4.2.1 below). However, as can be seen from the Hungarian data, not all instances of possessive marking are simply reducible to case marking.

### 3.3.3 *Other typological distinctions among possessive phrases*

This section details other typological features that can be used to distinguish different classes of possessive constructions. These include the syntactic status of the possessor (section 3.3.3.1) and the existence of a class of inalienably possessed nouns.

#### 3.3.3.1 *Determiner-like vs. modifier-like possessors*

Within the linguistic literature on adnominal possession, a typological distinction is often drawn between possessors that function more like determiners of the possessum and possessors that function more like modifiers (see Plank (1992)). All possessors—in unmarked and marked constructions alike—fall into one of these two categories.

**Determiner-like possessors** Possessors acting like determiners can be distinguished by several characteristics. First, the possessum cannot have any other determiners in phrases where a determiner-like possessive appears (Plank, 1992, 453). The English *'s*-marked possessor is a good example of a determiner-like possessor:

- (43) Pat’s book  
 \*the Pat’s book

Second, determiner-like possessors lead to a definite interpretation of the possessum. This is again illustrated with English (Plank, 1992, 453):

- (44) There’s a book on the table.  
 ?There’s Pat’s book on the table.

Lastly, modifiers must attach to the possessum before determiner-like possessors do. Modifiers, when they appear, will be adjacent to the possessum, not separated from it by the possessor, as seen in the following English example:

- (45) Pat's boring book  
 \*boring Pat's book

**Modifier-like possessors** Each of the three tests that distinguish determiner-like possessors above can also be applied in reverse to distinguish modifier-like possessors. First, the possessum can have both a determiner and a modifier-like possessor (Plank, 1992, 453). This can be seen in the following example from Ancient Greek:

- (46) he:                    to                    patròs                    oikía  
 the.F.SG.NOM the.M.SG.GEN father(M).SG.GEN house(F)SG.NOM  
 'the father's house' [grc] (Goodwin, 1894)

Second, modifier-like possessors do not imply anything about the definiteness of the possessum, as can be seen in the following example from Italian:

- (47) a. la            sua            casa  
 the.FEM 3SG.POSS house(FEM)  
 'her or his house' [ita]
- b. una    sua            casa  
 a.FEM 3SG.POSS house(FEM)  
 'a house of hers or his' [ita] (Plank, 1992, 453)

Lastly, since possessors function like modifiers in these constructions, the possessor may appear between the possessum and an adjective that modifies the possessum. This is demonstrated in the following example from Egyptian Arabic, where the modifier of the possessum appears at the right edge of the phrase, separated from its modificand by the possessor:

- (48) a. rubAAT    gazma            9AsmAr  
 lacing.M.SG shoe.pair.SG.F black.M.SG  
 'the black laces of the shoes' [arz] (note that the lacing is black, not the shoes)  
 (Mitchell, 1962, 49)

In some languages, no determiners exist. This means that one or more of these tests may not be applicable. In these cases, the relative ordering of modifiers and possessors may be the only evidence bearing on the analysis.

### 3.3.3.2 *Inalienable possession*

The term ‘inalienable possession’ refers to a phenomenon wherein some languages require that certain words appear with a possessor, or where a certain class of nouns take a unique possessive marking pattern that is not permitted on other nouns (Nichols, 1988, 563). Though there is no single unifying cross-linguistic meaning for inalienably possessed noun phrases, often inalienably possessed nouns are body part terms or kin terms, convey part-to-whole relations, or denote “culturally basic possessed items (e.g. arrows, domestic animals)” (Nichols, 1988, 572).

Navajo is an example of language where certain nouns are required to appear with possessors. This can be seen in the following examples, where the inalienably possessed noun *-be* ‘milk’ must appear with an overt possessor:

- (49) a. *bi-be*  
           3SG.POSS-milk  
           ‘her (own) milk’ (referring to mother’s milk) [nav]
- b. *’a-be*  
           3UNSP.POSS-milk  
           ‘(someone’s, something’s) milk’ [nav] (Nichols, 1988, 564)

In (49b), a ‘dummy’ possessor must be introduced so that the noun *-be* ‘milk’ does not appear without a possessor.

In Eastern Pomo, inalienable possession takes a slightly different form. Here, inalienably possessed nouns appear in possessum-marking constructions, and alienably possessed nouns appear in possessor-marking constructions:

- (50) a. Inalienable possession:  
           *wí-bayle*  
           1SG.POSS-husband

‘my husband’ [peb]

b. Alienable possession:

wáx        šári  
1SG.POSS basket

‘my basket’ [peb] (Nichols, 1988, 566)

Haspelmath has noted that, in languages with a distinction between alienable and inalienable possessive constructions, if “one of the constructions is overtly coded while the other is zero coded, it is always the inalienable construction that is zero-coded, while the alienable construction is overtly coded” (Haspelmath, 2017, 199). Haiman (1983, 1985) argues that this is a case of iconicity, where an inalienably possessed noun should be at as small as possible a ‘linguistic distance’ from its possessor. Haspelmath argues that this is more likely to be a result of frequency effects, since inalienably possessed nouns are almost always frequent lexical items (Haspelmath, 2017, 193). In general, inalienably possessed nouns are a small, closed class in all languages that draw this distinction.

For the purposes of this thesis, the alienable/inalienable distinction is not considered to be within the primary scope of work. In many cases, preexisting architecture from other libraries will allow the user to model this distinction. For example, the fact that inalienably possessed words must appear with a possessor pronoun in the form of an affix in Eastern Pomo (see (50a) above) can be modeled in the following manner: a user can create a position class for the possessor pronoun affix on the Morphology subpage of the questionnaire, indicate that it appears only on a certain class of nouns, and then make it obligatory in those cases where it appears. All these functions already exist within the Morphology subpage of the Grammar Matrix customization questionnaire, and nothing need be added to model this phenomenon. However, cases do exist where the current architecture does not suffice to model an alienable/inalienable distinction. This library does not extend the implementation to cover these cases.

### **3.4 Summary**

In this chapter, I have presented the various types of adnominal possessive phrases that are attested in the world's languages. This typological space is broken down by a number of features, including the number and location of possessive markers, the status of those markers, possessive phrase ordering, and the syntactic status of the possessor. Other phenomena, including agreement between possessor and possessum and inalienable possession, were also discussed. The following chapter is concerned with the analysis that I developed and subsequently implemented in order to model these phenomena.

## Chapter 4

# ANALYSIS

In this chapter, I outline an HPSG analysis of adnominal possessive phrases that covers the phenomena described in the previous chapter. Given the ultimate goal of implementing a Grammar Matrix library, I approach the present analysis with the aim of being able to model any of the constructions seen in chapter 3 based on a minimal amount of information from the user-linguist. To that end, in this chapter, I lay out this analysis in terms of a few binary- or ternary-valued features that distinguish possible types of adnominal possessive phrases and define the boundaries of the typological space under discussion (e.g. the location of a possessive marker or the form the marker takes). I also strive to select analyses for individual phenomena that are maximally general, rather than adapted only to a single language or cluster of features. The ability to capture what is true, say, of all modifier-like constructions, or of all unmarked constructions serves the purpose of capturing potentially significant generalizations, as well as making the resulting analysis more readily implementable in the form of a Grammar Matrix library.

The present chapter is organized as follows. First, in section 4.1 I give the target semantic representation that any possessive phrase should map to. Next, I step through the analyses that allow one to map possessive phrases to these target semantics. This portion of the chapter is organized by the features that factor the typological space. The primary division between sections is the syntactic status of the possessor, with all modifier-like possessors covered in section 4.2 and all specifier-like possessors covered in section 4.3. Within each of these sections, I distinguish between phrases with zero, one, or two possessive markers. I further distinguish between markers that take the form of affixes and those that take the form of separate words. Finally, I discuss the extension of this analysis to cover possessor

pronouns (section 4.4), as well as agreement between possessor and possessum (section 4.5). Note that throughout this discussion, it is assumed that more than one possessive strategy or set of possessor pronouns may be implemented by the user-linguist in a given grammar. This analysis must account for potential interactions between these possessive strategies and pronouns.

#### 4.1 Semantic representation

Following materials for the University of Washington Linguistics 567 course<sup>1</sup> and the English Resource Grammar (Flickinger, 2000, 2011), I encode possession in the Minimal Recursion Semantics formalism by means of a *poss\_rel* relation between possessor and possessum, as shown in section 3.2.2 above. This can be seen in the MRS representation of the English sentence *The dog's cat sleeps* given in (51):

$$(51) \left[ \begin{array}{l} mrs \\ LTOP \ h14 \\ INDEX \ e1 \\ \\ \left. \begin{array}{l} \left[ \begin{array}{l} PRED \ exist\_q\_rel \\ LBL \ h8 \\ ARG0 \ x2 \\ RSTR \ h5 \\ BODY \ h9 \end{array} \right], \left[ \begin{array}{l} PRED \ \_dog\_n\_rel \\ LBL \ h6 \\ ARG0 \ x3 \end{array} \right], \left[ \begin{array}{l} PRED \ poss\_rel \\ LBL \ h4 \\ ARG0 \ e10 \\ ARG1 \ x2 \\ ARG2 \ x3 \end{array} \right], \\ \left[ \begin{array}{l} PRED \ exist\_q\_rel \\ LBL \ h11 \\ ARG0 \ x2 \\ RSTR \ h7 \\ BODY \ h12 \end{array} \right], \left[ \begin{array}{l} PRED \ \_cat\_n\_rel \\ LBL \ h4 \\ ARG0 \ x2 \end{array} \right], \left[ \begin{array}{l} PRED \ \_sleep\_v\_rel \\ LBL \ h13 \\ ARG0 \ e1 \\ ARG1 \ x2 \end{array} \right] \end{array} \right\} \\ \\ \left. \begin{array}{l} \left[ \begin{array}{l} qeq \\ HARG \ h5 \\ LARG \ h6 \end{array} \right], \left[ \begin{array}{l} qeq \\ HARG \ h7 \\ LARG \ h4 \end{array} \right] \end{array} \right\} \end{array} \right]$$

<sup>1</sup><http://courses.washington.edu/ling567/2017/lab5.html#poss>

This analysis is desirable because it serves as a relatively underspecified way to represent a range of relationships that can exist between possessor and possessum. The two are linked as follows: the ARG0 value of the possessum is identified with the ARG1 value of the *poss\_rel*, while the ARG0 value of the possessor is identified with ARG2 of the *poss\_rel*. In addition, the LBL value of the possessive relation is shared with the LBL of the possessum, because both relations have strictly the same scope.

The *poss\_rel* must be added by some piece of the grammar that is specific to possessive constructions. This may be a binary phrase rule, a unary phrase rule, an inflectional rule, or a lexical entry. In any case, the *poss\_rel*-adding element must have access to the INDEX values of both the possessor and the possessum, as well as the LTOP value of the possessum. In general, in formulating this analysis, I have striven to ensure that *poss\_rel* is added by a rule or lexical entry that has phonological content; it is usually not added by a zero-inflecting rule or a phrase rule that does not correspond to some phonological material. Of course, in the case of constructions where there is no overt marking of possession, it is impossible to attach the *poss\_rel* to a morphological marker of possession, and so the *poss\_rel* is instead carried by the binary phrase rule that combines possessor and possessum.

One exception to this rule of the *poss\_rel* being linked to overt markers of possession is the case where the possessum is marked and the possessor is specifier-like. In this case, the semantics are carried by a unary phrase rule that attaches to the possessor. The reasons for this are discussed in section 4.3.2.2 below.

In addition to the *poss\_rel*, specifier-like possessors must contribute the quantifier for the possessum. These possessors take the place of the determiners that would ordinarily contribute these quantifiers and so they must play this role. Modifier-like possessors do not take the place of determiners and so do not contribute this quantifier.

## **4.2 Modifier-like possessive constructions**

The present section is concerned with describing an analysis for modifier-like possessors that captures as many generalizations about these constructions as possible, while being adaptable

to the various subtypes of modifier-like constructions. The features that are used to factor the typological space are as follows: the number (zero, one, or two), location (possessor or possessum), and type (affix or non-affix) of markings.

Modifier-like possessors are distinguished from specifier-like possessors by the fact that any determiner-taking possessum may have both a modifier-like possessor and a determiner and that modifier-like possessors may occur in any order relative to other modifiers of the possessum. In addition, the modifier-like possessor does not require a definite reading for the possessum. A full description of this distinction is given in section 3.3.3.1 above. In this section, I detail the analysis for unmarked modifier-like constructions in subsection 4.2.1, and the analysis for marked modifier-like constructions in subsection 4.2.2.

The major points of difference between any given modifier-like construction and its specifier-like counterpart are the following: first, while in modifier-like constructions, the phrase consisting of possessor and possessum has a non-saturated SPR value, in specifier-like constructions, the possessor + possessum complex always has the value [ SPR  $\langle \rangle$  ]. This means that in languages with determiners, in the modifier-like case, the possessor + possessum phrase will still combine with a determiner. Second, because the possessum has an unsaturated SPR value in the modifier-like case, it should remain unquantified until it combines with a determiner or passes through the *bare-*np** rule, either of which contributes a quantifier. Therefore, the possessor does not contribute any quantifier for the possessum in these constructions.

#### 4.2.1 *Unmarked possessive phrases*

Unmarked possessive constructions differ dramatically from marked constructions in that no morphological material is introduced which can carry the necessary semantic information to build a possessive phrase, particularly the *poss\_rel*. For this reason, a new binary phrase type must be introduced, called *poss-phrase*. This phrase rule is shown as an AVM in (63) below. It has as its supertype either *head-initial* or *head-final* (both of which are defined in `matrix.tdl`) as appropriate to generate the desired ordering.



for a specifier of some description. The parent of this rule, then, is [ SPR  $\langle X \rangle$  ], meaning that the head-daughter's requirement for a specifier has not been satisfied.

Both daughters and the parent of this rule are subject to several constraints. First, the head daughter cannot be a pronoun, insofar as phrases like *\*my he* or *\*Pat's us* are disallowed in all languages. This constraint is encoded by requiring that the head daughter be [ PRON – ]. The non-head daughter may also be optionally specified to be [ PRON – ], since languages with unmarked possessive constructions may or may not allow pronouns to act as the possessor in these constructions.<sup>2</sup>

Furthermore, both the head and the non-head daughter are constrained by the features [ POSSESSOR *nonpossessive* ] and [ POSSESSUM *nonpossessive* ]. The POSSESSOR and POSSESSUM features are used in strategies where the possessor and/or the possessum is marked, and is described in greater detail in sections 4.3.2 and 4.2.2 below. These constraints are included on this rule for those (very common) cases where more than one possessive strategy is implemented in a given language, and a marked possessor or possessum must be prevented from appearing in this phrase type, which is intended only for unmarked nouns.

In addition to these automatically included constraints, other constraints on the possessor may optionally be added by the user-linguist. One clear use case for these user-specifiable constraints on the possessor is the recommended analysis for languages where the possessor is marked by the genitive case. Simply having the genitive case affix carry the possessive semantics would mean that the genitive case could only be used in adnominal possessive constructions, while many languages use the genitive case in a host of other contexts as well. As an alternative analysis, the genitive case affix could simply be homophonous with but distinct from the possessor marking—an analysis which seems more contrived than would be ideal. Instead, I have recommended (through the documentation page) that the user analyze the possessive construction as an unmarked construction wherein the possessor is

---

<sup>2</sup>The Grammar Matrix did not previously include the PRON feature for marking pronouns as distinct from other nouns. The option of adding this HEAD feature to noun types was added in the questionnaire in order to rule out pronouns appearing in the possessum role.

constrained to have [ CASE *gen* ]. This allows the genitive case affix to be identical to the affix that appears on the possessor without needing to carry the possessive semantics into contexts where they would not be appropriate.

Just as the user can specify additional constraints on the possessor, it would be possible to allow the user to add similar constraints to the possessum. However, this option is not currently implemented in the library. This is because it is assumed that, for the most part, if the possessum (being the head of the possessive phrase) is subject to constraints such as case, these constraints come from other elements of the sentence outside the possessive phrase. For example, in a nominative-accusative language, a possessive phrase in object position should see the possessum marked with accusative case, not because it is the head of a possessive phrase, but because it is the complement of a transitive verb. Therefore, feature constraints on the possessum are assumed to be the product of other elements of the Grammar Matrix customization system and not of constraints on this rule. This is true of the possessum in the unmarked constructions that are the subject of this current section, but it extends to all other constructions as well. One known possible use case for user-specifiable possessum feature constraints is modeling inalienable possession (see section 6.2.4). Since this phenomenon is not within the scope of the present thesis, this is not implemented.

In summary, unmarked modifier-like possessive phrases are modeled by a phrase structure rule that adds the necessary semantics, and which takes a nominal, [ SPR ⟨ ⟩ ] non-head daughter and a nominal, [ SPR ⟨ *X* ⟩ ] head daughter, and in turn produces a phrase which is [ SPR ⟨ *X* ⟩ ]. Various constraints on the head daughter and non-head daughter are added (or can optionally be added by the user) to ensure that only the correct nouns appear in these constructions.

#### 4.2.2 *Marked possessive phrases*

In contrast to the analysis of unmarked modifier-like possessive constructions, marked possessives can take advantage of existing phrase structure rule types. In these constructions, the possessive phrase is analyzed as an instance of one of two rules: a head modifier rule

(called *head-adj-int* in the Grammar Matrix, an abbreviation for ‘head-adjunct-intersective’), or a head-complement (*head-comp*) rule.<sup>3</sup> In this section, I discuss the choice of phrase rules used to model modifier-like constructions, solutions to ordering issues, the rationale for possessive features, and the use of CASE to limit overgeneration. I then discuss variations in the analysis based on the whether there is a marking on the possessor (section 4.2.2.1), the possessum (section 4.2.2.2), or on both (section 4.2.2.3).

**Phrase structure rule selection** The choice of whether to use *head-comp* or *head-adj-int* rules to model possessive phrases depends on the location of the possessive marking. In general, my preferred analysis is not to locate the *poss\_rel* on an unmarked constituent when marked constituents exist. For that reason, the *poss\_rel* would be located on the possessor when the possessor is marked, the possessum when it is marked, and on either one when both are marked. Note, however, that the constituent bearing the *poss\_rel* must have access to the INDEX of both the possessum and the possessor, in order to identify them with its ARG1 and ARG2 respectively. This is not problematic in head-specifier constructions, since these constructions are mutually selecting, with the head selecting for its specifier via the SPR list, and the specifier selecting for its head via the SPEC list. Whether the *poss\_rel* is located on the possessor or the possessum, it will have access to the INDEX values of both.

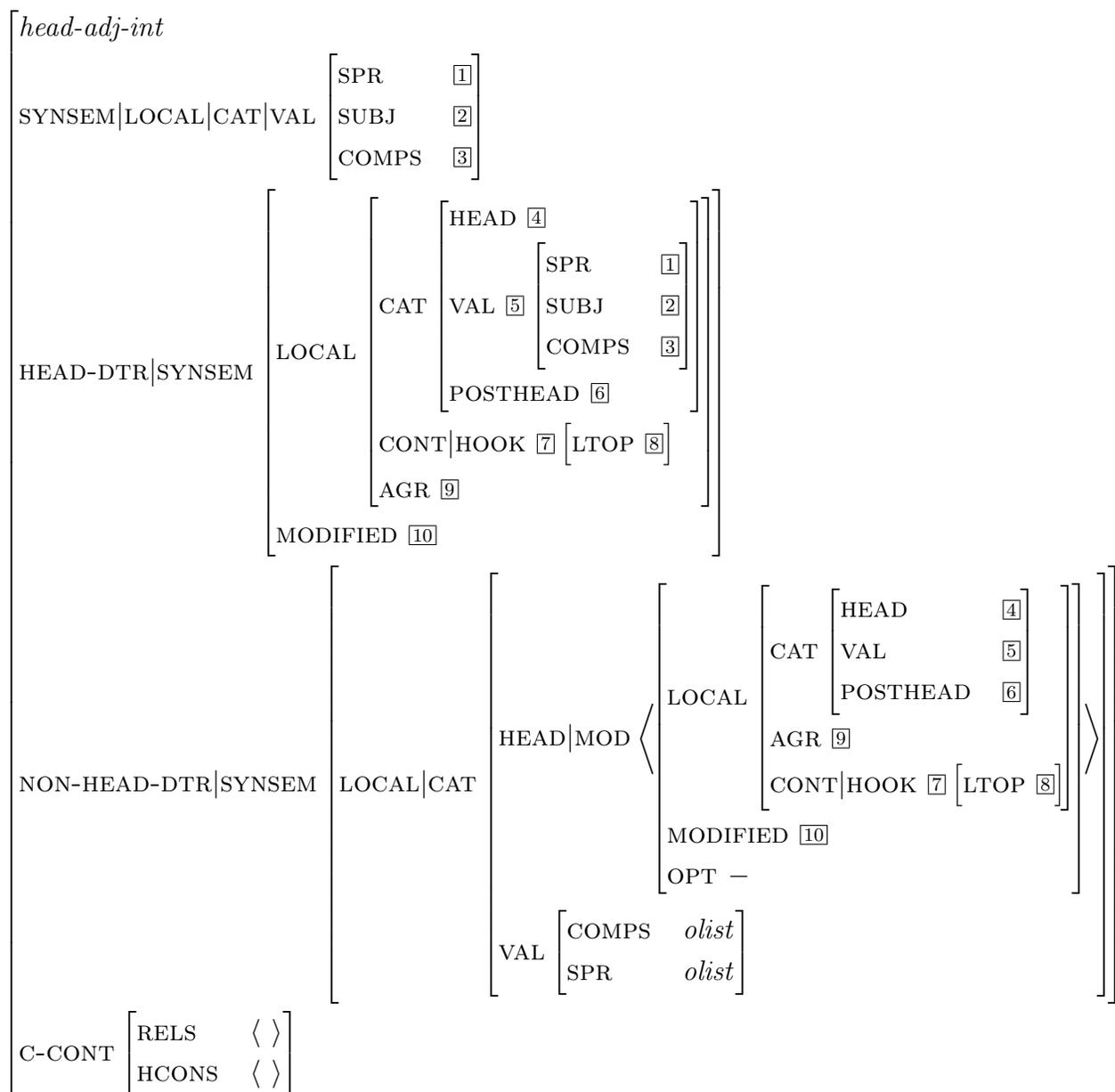
Problems arise with head-adjunct constructions, because the adjunct selects for its head, but not vice versa. If the marking (and the *poss\_rel* that accompanies it) is on the adjunct (that is, the possessor), then the *poss\_rel* can correctly identify its ARG values with the INDEX values of the possessor and the possessum. However, if the marking and the *poss\_rel* appear on the head (that is, the possessum), it will not be able to identify its ARG2 value with the INDEX of the possessor. Therefore, given that my goal is to preserve the generalization that the *poss\_rel* is added by a marked constituent, it is necessary to use *head-comp* to model those constructions where the possessum is marked and the possessor is modifier-

---

<sup>3</sup>Note that, since both the *head-comp* and the *head-adj-int* rules in the Grammar Matrix are already head-compositional, it is not necessary to add any unary rule of the sort used with specifier-like possessors in section 4.3.2 below in order to sort out semantics. Instead, the possessor and possessum are combined directly by the appropriate binary phrase structure rule.

like.<sup>4</sup> In *head-comp*, it is the head that selects for the complement, giving it access to both constituents' INDEX values. Therefore, I use *head-adj-int* (see (53)) to model modifier-like possessor-marking constructions and *head-comp* to model modifier-like possessum-marking constructions.

(53)



<sup>4</sup>For example, Hungarian (see (37))

In (53), the *head-adj-int* rule is given, with all relevant features shown, including constraints that appear on the supertypes of *head-adj-int*, such as *isect-mod-phrase*, *basic-head-mod-phrase-simple*, and *head-adj-phrase*. These are all defined in `matrix.tdl`. Other super-types of *head-adj-int* whose constraints are not shown below include *binary-headed-phrase* and *head-mod-phrase*.

The head-complement rule from the Grammar Matrix is shown in (54) below. This type definition includes some relevant constraints from its immediate supertype *basic-head-comp-phrase*. This type also inherits from the rules *head-valence-phrase*, *head-compositional*, and *binary-headed-phrase*, as defined in `matrix.tdl`.

$$(54) \left[ \begin{array}{l} \textit{basic-head-1st-comp-phrase} \\ \text{SYNSEM} \left[ \begin{array}{l} \text{LOCAL.CAT} \left[ \begin{array}{l} \text{VAL} \left[ \begin{array}{l} \text{COMPS} \quad \boxed{1} \\ \text{SUBJ} \quad \quad \boxed{2} \\ \text{SPEC} \quad \quad \boxed{3} \\ \text{SPR} \quad \quad \quad \boxed{4} \end{array} \right] \end{array} \right] \end{array} \right] \\ \text{HEAD-DTR.SYNSEM} \left[ \begin{array}{l} \text{LOCAL.CAT} \left[ \begin{array}{l} \text{VAL} \left[ \begin{array}{l} \text{COMPS} \quad \langle \boxed{5}, \boxed{1} \rangle \\ \text{SUBJ} \quad \quad \boxed{2} \\ \text{SPEC} \quad \quad \boxed{3} \\ \text{SPR} \quad \quad \quad \boxed{4} \end{array} \right] \end{array} \right] \end{array} \right] \\ \text{NON-HEAD-DTR.SYNSEM} \quad \boxed{5} \\ \text{C-CONT} \left[ \begin{array}{l} \text{RELS} \langle \ \rangle \\ \text{HCONS} \langle \ \rangle \end{array} \right] \end{array} \right]$$

**Addressing order variation** Problems with phrase ordering may arise in modifier-like constructions. As there is no guarantee that the ordering of preexisting *head-adj-int* or *head-comp* rules will correspond to the ordering of possessor and possessum in a given language.

This issue of distinctive orderings for certain subtypes of major phrase rules is already dealt with in the clausal complements library (Zamaraeva and Howell, To be presented), where different complementizers may require head-initial ordering or head-final ordering

within a single language. In order to model this behavior, Zamaraeva and Howell (To be presented) add a feature, `INIT`, to `HEAD` within the clausal complements library. Complementizers that must precede their complements are constrained to be `[ INIT + ]`, while those that follow their complements are `[ INIT - ]`. The (head-initial) *head-comp* rule is then required that its head daughter be `[ INIT + ]`, while the comp-head rule imposes the constraint that its head daughter be `[ INIT - ]`. This precise approach can be employed in cases where the possessive phrase is a *head-comp* phrase.

As for *head-adj-int* constructions, the adjectives library (Trimble, 2014) has a means to deal with the fact that some adjectives require a specific ordering relative to the noun they modify. Much like the feature `INIT`, the `POSTHEAD` has the value `+` on adjectives that appear after the nouns they modify, and the value `-` on adjectives that appear before the nouns they modify. Adjectives which can appear with either ordering are left unconstrained for this feature. The variously ordered instantiations of *head-adj-int* are then constrained to require a `[ POSTHEAD + ]` or `[ POSTHEAD - ]` non-head daughter, depending on their own ordering. Both these features are used to appropriately constrain the ordering of possessive phrases.

**Possessive features** Finally, it is important to note that in all languages with marked constructions, it is necessary to have some means of ensuring that unmarked nouns do not appear in possessive constructions. In some cases, it would be theoretically possible to achieve this end by simply constraining the `VAL` and `MOD` features of nouns. For example, if all unmarked nouns have empty `MOD` lists, then simply adding a non-empty `MOD` list to marked possessors guarantees that only marked possessors will appear in possessive constructions. However, this approach breaks down in more complex constructions, particularly constructions where the possessor is modifier-like. In order to model all constructions, possessive-marking features must be introduced to distinguish between marked and unmarked nouns.

The features created to help model possession are `POSSESSOR`, which resides at `HEAD`, and `POSSESSUM`, which resides at `CAT`. These are kept as two distinct features in order to ensure that they are propagated—or prevented from propagating—through the tree as appropriate. In the case of `POSSESSUM`, the feature value must not be propagated up from

the possessum by headed phrase structure rules, as a HEAD feature always is when it appears on the head daughter (keeping in mind, of course, that the possessum is always the head daughter of a possessive phrase). If the POSSESSUM feature were to be propagated upwards through the parse tree, it would essentially mark any possessive phrase as though it were itself a possessum, rather than simply containing one. Possessive noun phrases are not expected to have any distributional differences from nonpossessive noun phrases, and so this would be incorrect. Therefore, POSSESSUM must not be in HEAD.<sup>5</sup> By contrast, the POSSESSOR feature never occurs on the head daughter of the possessive phrase, and so it does not pose this issue of propagating throughout the parse tree inappropriately. Furthermore, keeping the POSSESSOR feature in HEAD is convenient for those cases where the possessor is marked by a separate word, rather than an affix. In these constructions, the marking word takes the possessor as its complement. The phrase consisting of the marker and the possessor should be marked as a possessor — that is, the marker should pass its POSSESSOR value up to the parent of this phrase, which happens automatically when it is a HEAD feature. For this reason, the POSSESSOR feature should be in HEAD, even while the POSSESSUM feature remains at CAT.

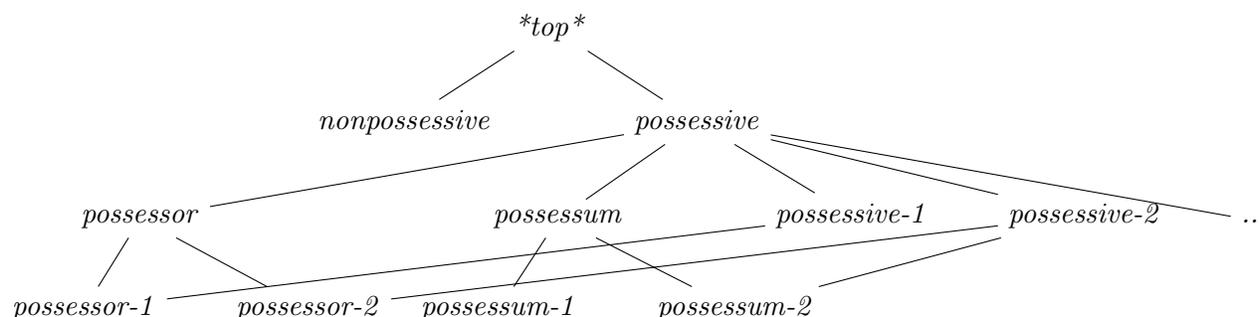
Both these features take values of the type *poss*. The type *poss* has the subtypes *possessor*, *possessum*, and *nonpossessive*. Furthermore, there must be subtypes of *possessor* and *possessum* in order to allow users to instantiate multiple possessive strategies, without the various parts of those strategies mixing in single constructions. To this end, *possessive* can have indefinitely many numbered subtypes, *possessive-1*, *possessive-2*, etc. Then, multiply

---

<sup>5</sup>Note that in order to make this analysis work, the head-complement rule is made to pass the value of POSSESSUM up the tree so that the complex of possessum-marker + possessum has the value [ POSSESSUM *possessum* ]. Likewise, after evaluation, it was determined that the head-modifier rule should pass up this value in the case where the possessive construction is a head-specifier construction. This allows a modifier to attach to the possessum without obscuring the fact that it is a marked possessum, which the possessor must be able to find out. This correction was made in post-evaluation editing (see section 6.4). However, no other phrase structure rules pass up this value. This ensures that the feature value [ POSSESSUM *possessum* ] marks the complex consisting of possessum-marker and possessum, but is dropped when this complex passes through the *bare-np* rule or acquires a determiner, and so will not propagate to the top level of the possessive phrase.

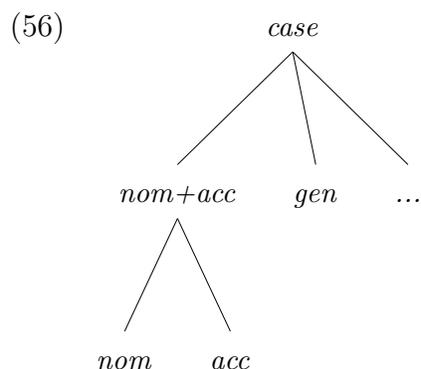
inheriting subtypes are created: *possessor-1*, *possessum-1*, *possessor-2*, *possessum-2*, etc. The feature hierarchy for *poss* is shown in (4.2.2) below:

(55)



**Using case to block overgeneration** Just as it is important to ensure that unmarked nouns do not appear in the role of marked possessive nouns, it is important to ensure that marked nouns do not appear outside of possessive phrases. A marked possessum is generally prevented from appearing without its possessor by having an unmet requirement for a SPR (or, as is outlined in section 4.2 below a COMP), which it must satisfy by combining with a possessor before it can appear in any other context. However, having a non-empty MOD or SPEC list does not prevent a marked possessor from appearing in all contexts. It would be possible to achieve this by simply requiring that all non-possessive argument-taking entities in the grammar required that their arguments have their POSSESSOR and POSSESSUM features set to *nonpossessive*. However, it is possible to achieve much the same effect by taking advantage of a system already in place in the Grammar Matrix: case. In languages that have a case system, the following analysis is employed in order to rule out, for example, a marked possessor acting as the argument of a verb.

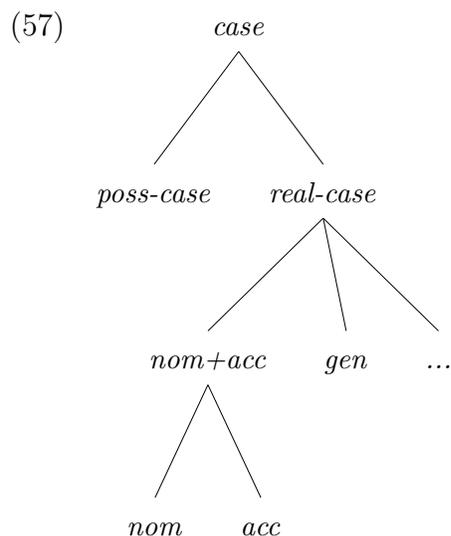
In the Grammar Matrix, when a user implements case for a language, all case types (e.g. *nominative*, *accusative*, *ergative*, *absolutive*) inherit from *case* either directly or indirectly, as shown in (56) below:



In this analysis (see (57)), all case types inherit from an intermediate type, called *real-case*. This *real-case* type contrasts with the dummy type *poss-case*, making *poss-case* incompatible with all the daughters of *real-case*. If case exists in the language and at least one possessive strategy has been implemented, then the analysis shown in (57) is implemented, and all marked possessors are constrained to be [ CASE *poss-case* ]. Then, the arguments of all verbs, auxiliaries, adjectives, and adpositions are constrained to be [ CASE *real-case* ]. In this way, an inflected possessor is prevented from appearing in any of these contexts without its possessum. Note that this is only strictly necessary in the case where the possessor marking takes the form of an affix. Possessor-marking adpositions also carry *poss-case*, and so, in languages where case-marking adpositions exist, blocking is achieved in a similar way. In those languages that do not have case-marking adpositions, the fact that the subphrase consisting of the possessor and its marking adposition has HEAD type *adp* prevents it from appearing in contexts appropriate for nouns.

This solution works automatically for those cases where the language has a case system. It can be used in languages without a case system as well, though at the cost of implementing a dummy case system with zero-inflecting case affixes, which would generate the correct *real-case* and *poss-case* features in order to keep possessors from appearing in the wrong contexts. However, this dummy case system is hard to justify from a theoretical standpoint. Fortunately, this case appears to be rare: most languages in Dryer and Haspelmath (2013) that have an affixally-marked possessor are analyzed as having a case system. Since there

appear to be few languages that have a possessor-marking inflection that are not analyzed as having a case system, the ‘dummy case’ solution suffices to cover this limited set of languages.



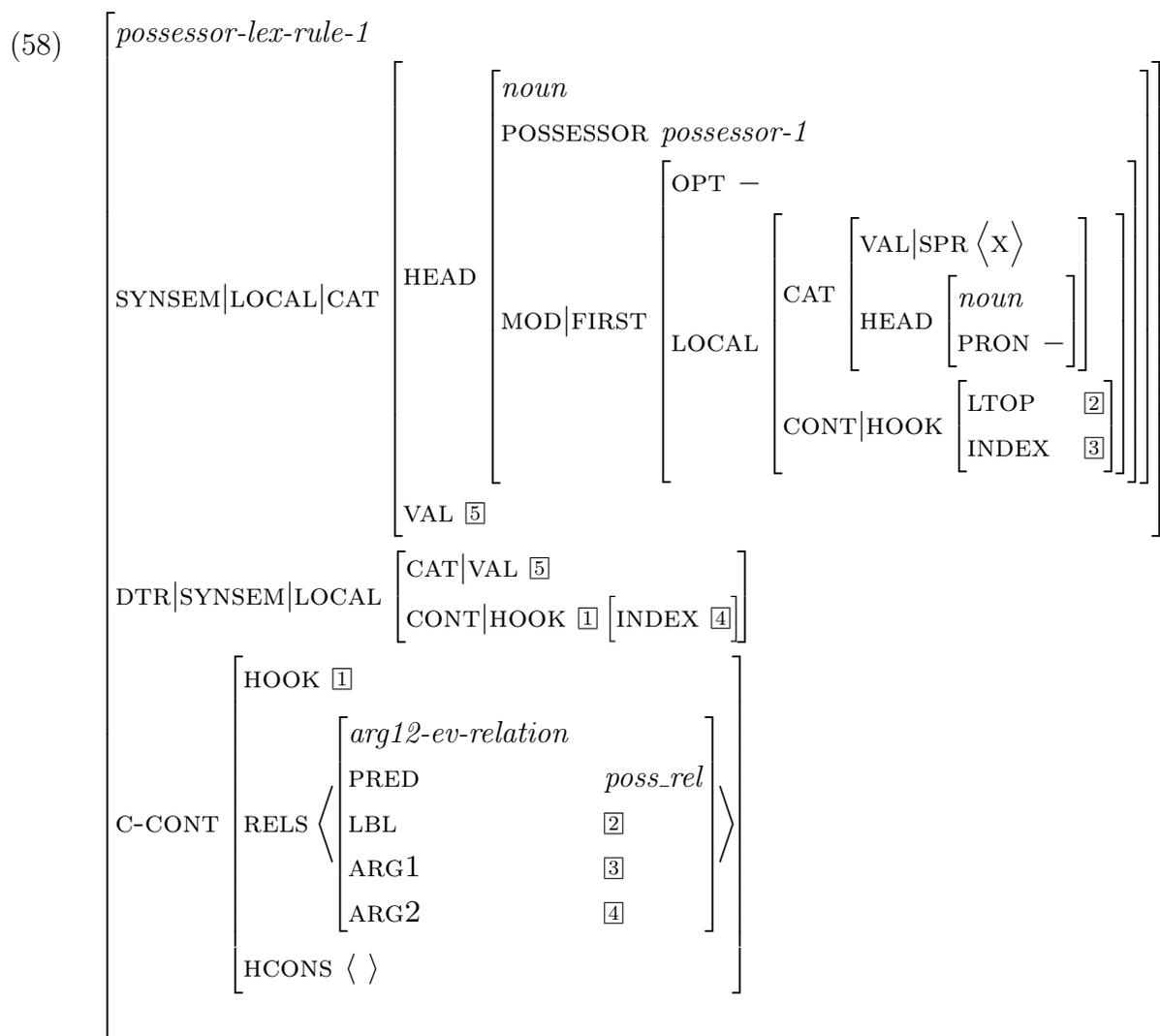
All marked possessive constructions with modifier-like possessors employ this analysis as recounted to this point, including the use of a unary phrase structure rule, the features POSSESSOR and POSSESSUM, the features INIT or POSTHEAD, and the case system (in cases of affixal marking). Now I turn to the specific extensions of this analysis in the case of various marking patterns.

#### 4.2.2.1 Possessor-marking

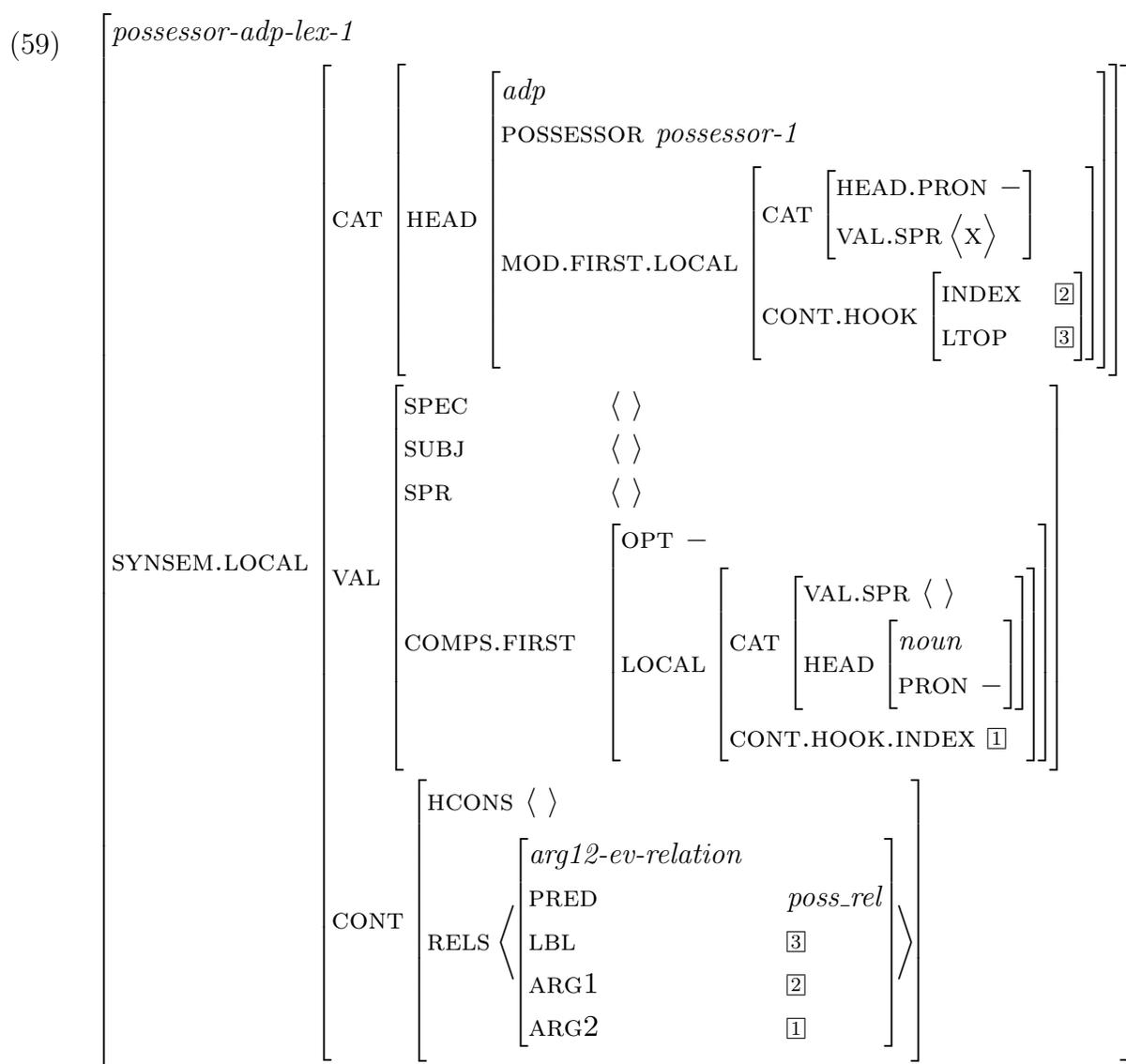
As discussed above, when a modifier-like construction has marking on the possessor, it is analyzed as an instance of *head-adj-int*. The possessor marking may take the form of either an affix or a separate word.

**Affixal marking** The lexical rule that adds the possessor-marking affix also adds the *poss\_rel*, as well as adding an item to the possessor’s MOD list, allowing it to act as the modifier of the possessum. Note that the item on the MOD list of this rule (that is, the possessum) does not have a saturated SPR value, and the possessor will not satisfy this need for a specifier. This allows for the possessum to (correctly) still combine with a specifier.

The lexical rule is shown in (58) below.



**Non-affixal marking** Any non-affixal marker that appears on the possessor (including clitics, which are analyzed as fully syntactically independent in the Grammar Matrix) is analyzed as an adposition that takes the possessor as its complement. Its lexical entry is shown in (59) below:



Other analyses for this marker are possible — for example, in the English Resource Grammar (Flickinger, 2000, 2011), the English clitic marker *'s* is analyzed as a determiner, which takes the possessor as its specifier, and then acts in turn as the specifier of the possessum. For the purposes of this library, the analysis of the marker as an adposition is preferred because it is adaptable to both the modifier-like and specifier-like scenario, whereas the determiner analysis is appropriate only for the specifier-like scenario. This broader applicability is desirable in the multilingual grammar engineering context. The possessor-marking adposition is also marked [ POSSESSOR *possessor* ]. This adposition has one item — the possessum — on

its MOD list, and as above the possessum is constrained to have an unsaturated SPR value.

#### 4.2.2.2 Possessum-marking

As discussed above, when the possessum is marked, the possessive phrase is analyzed as a head-complement phrase. The marked possessum therefore carries an item (corresponding the possessor) on its COMPS list.

**Affixal marking** The possessum-marking affix adds an item to the possessum’s COMPS list. It requires that this item — the possessor — be a full noun phrase with a saturated SPR value. The AVM for the possessum-marking affix is given here:

$$(60) \left[ \begin{array}{l} \textit{possessum-lex-rule-1} \\ \\ \text{SYNSEM|LOCAL|CAT} \left[ \begin{array}{l} \text{HEAD} \left[ \begin{array}{l} \textit{noun} \\ \text{POSSESSOR } \textit{nonpossessive} \end{array} \right] \\ \text{POSSESSUM } \textit{possessum-1} \\ \text{VAL} \left[ \begin{array}{l} \text{SPR } \boxed{5} \\ \text{COMPS|FIRST} \left[ \begin{array}{l} \text{OPT } - \\ \text{LOCAL} \left[ \begin{array}{l} \text{CAT} \left[ \begin{array}{l} \textit{cat-sat} \\ \text{VAL|SPR } \langle \rangle \end{array} \right] \\ \text{CONT|HOOK } \left[ \text{INDEX } \boxed{3} \right] \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right] \\ \\ \text{DTR|SYNSEM|LOCAL} \left[ \begin{array}{l} \text{CAT} \left[ \text{VAL|SPR } \boxed{5} \right] \\ \text{CONT|HOOK } \boxed{1} \left[ \begin{array}{l} \text{INDEX } \boxed{4} \\ \text{LTOP } \boxed{2} \end{array} \right] \end{array} \right] \\ \\ \text{C-CONT} \left[ \begin{array}{l} \text{HOOK } \boxed{1} \\ \text{RELS} \left\langle \begin{array}{l} \textit{arg12-ev-relation} \\ \text{PRED} \quad \textit{poss\_rel} \\ \text{LBL} \quad \boxed{2} \\ \text{ARG1} \quad \boxed{3} \\ \text{ARG2} \quad \boxed{4} \end{array} \right\rangle \\ \text{HCONS } \langle \rangle \end{array} \right] \end{array} \right]$$

**Non-affixal marking** A non-affixal marker on a possessum is analyzed as a noun that

takes the possessum as its complement. This is the case in both the modifier-like constructions under discussion here and the specifier-like constructions discussed in section 4.3.2 below. While it would be more consistent to analyze these markers as adpositions, as in the case of the possessor marker, the possessum marker is ultimately the head of the possessive phrase, which must be headed by a noun, not an adposition. The possessum-marking noun (shown in (73)) is essentially identical to the possessor-marking adposition, save for the fact that it is of type [ HEAD *noun* ], rather than [ HEAD *adp* ]. The marker identifies the HOOK value of its complement with its own HOOK value. This ensures that the INDEX value of the possessum is identified with the INDEX value of the possessive phrase overall.

In this case — where a possessum takes a non-affixal marking in a modifier-like construction — this analysis makes a prediction about the order in which the possessum-marking word and the determiner attach to the possessum. The *poss\_rel*, which is located on the possessum-marking noun, must have the same scope as the possessum, which is modeled by having the *poss\_rel* and the possessum’s core relation share a LBL value. In order for the possessum-marking noun to identify the LBL value of the *poss\_rel* with the LBL value of the possessum’s core relation, it must have access to both. In general, nouns make the LBL value of their core relations accessible to other entities in the grammar by identifying it with a feature called LTOP that is located at HOOK. However, this LTOP value is not passed up the tree by the *head-spec* rule, which passes up the LTOP value of the non-head daughter instead. This means that once a determiner attaches to the possessum, the noun’s LTOP value is lost and not accessible to other entities in the grammar. For this reason, this analysis predicts that the marking word must attach to the possessum *before* the possessum attaches to any determiner it may require (or, equivalently, before it passes through the *bare-np* rule that adds a quantifier).<sup>6</sup> This attachment order is a prediction of this analysis which should be verified empirically.

---

<sup>6</sup>Note that in the specifier-like case, the possessum’s determiner slot is filled by the possessor, which attaches after the possessum marker, making this a non-issue. Therefore, the attachment order in the specifier-like case is determiner first, then possessor-marking word.

However, I have as yet found only one example of a language that can be analyzed as having this particular set of features—Fijian—and it does not provide evidence for or against this analysis. In Fijian, one of the possible possessive strategies attested is analyzed by Dixon (1988) as a possessum marking morpheme taking the form of a clitic<sup>7</sup> (which has the syntactic status of a separate word), with the possessor acting as a modifier:

- (61) a    *liga=i*            *Jone*  
           the hand=POSS John  
           ‘John’s hand’ [fij] (Dixon, 1988, 120)

However, since the determiner attaches from the left and the possessum-marking clitic—according to Dixon’s analysis—attaches from the right, it is impossible to tell from this construction what the attachment order is. What is more, as detailed in section 6.2.4 below, I settle on an analysis of Fijian where the possessive marking clitic is syntactically a dependent of the possessor, and phonologically a dependent of the possessum, which means that in my analysis, Fijian fails to provide any evidence about this scenario. Therefore, this attachment order—possessum-marking word, then determiner—is implemented in the library, and given as a prediction of this analysis, subject to confirmation or refutation based on evidence that has, as yet, not been found.

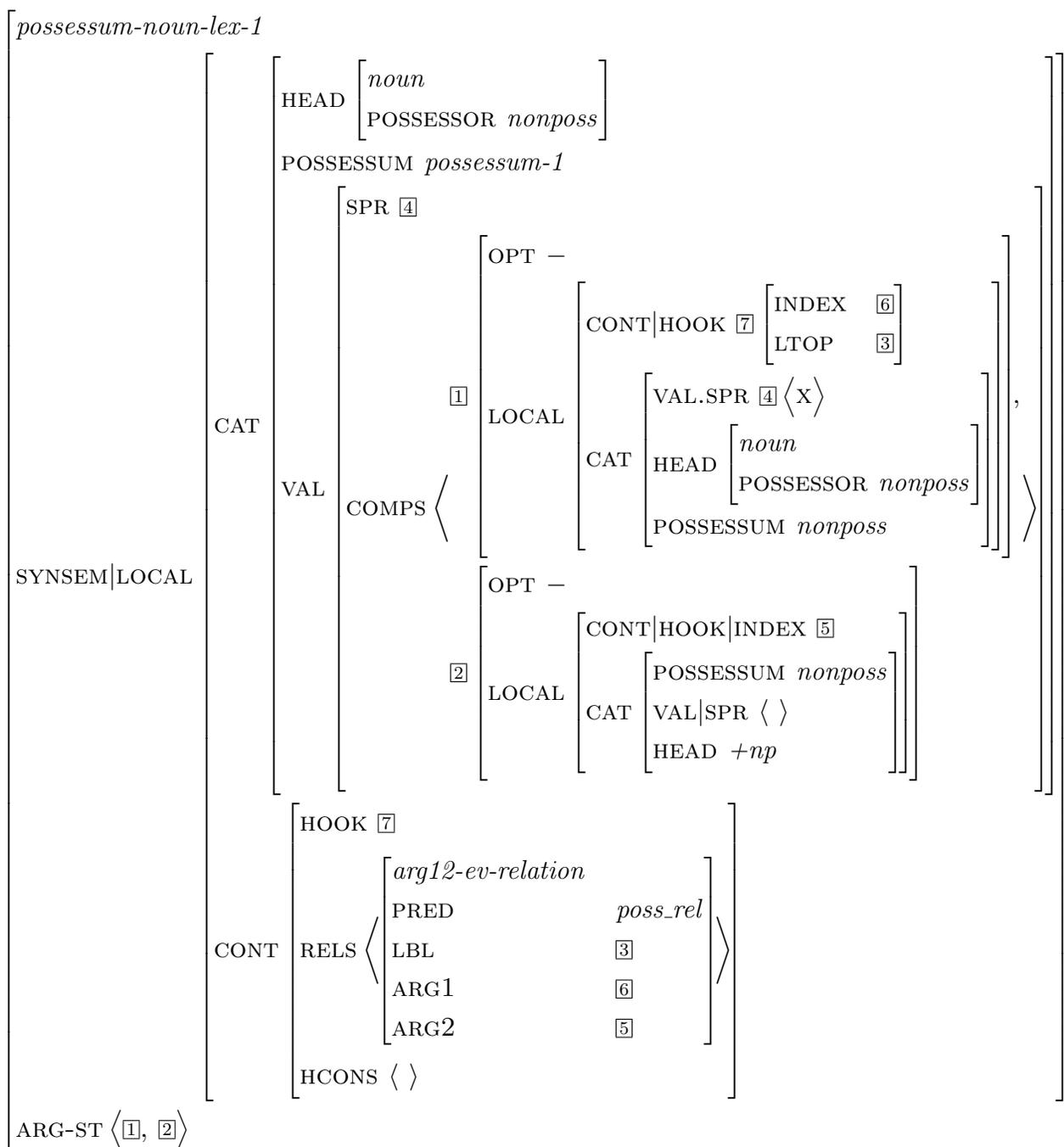
As was the case with the possessum-marking affix, the possessum-marking word takes the possessor as its complement. However, it first takes the possessum as its complement, since in general, possessum- and possessor-marking words are analyzed as taking the constituent they mark as a complement. The lexical entry for the possessum-marking word is shown below in (62).<sup>8</sup>

---

<sup>7</sup>Dixon (1988) actually calls this morpheme an affix, but provides evidence that clearly suggests that it is a clitic. See section 6.2.4 for details.

<sup>8</sup>Due to space constraints, in this AVM the feature value *nonpossessive* is abbreviated as *nonposs*.

(62)



Note the fact that both the possessum and the possessor are found on the COMPS list, in that order. The possessum noun itself should be unmarked as a possessive, while the

possessor may or may not be marked as a possessor, depending on the language. In (62), the possessor's value for POSSESSOR is left unconstrained, but it could be constrained to be marked by giving it the value [ POSSESSOR *possessor-n* ], where *n* is the appropriate number for the given strategy.

#### 4.2.2.3 *Both-marking*

Since possessor- and possessum-marking modifier-like constructions take different major phrase types, double-marking constructions must use one or the other of these phrase types. Since both constituents are marked, the *poss\_rel* can be carried by either one, so there is no strong argument in favor of one over the other. In the library, double-marking constructions were implemented as instances of *head-comp* phrases, but either version is possible.

#### 4.2.3 *Summary*

In this section, I have given an analysis for modifier-like possessive constructions. These constructions can be parameterized in terms of the number of overt markings of possession (zero, one, or two), the location of marking (possessor or possessum), and the type of the marker in question (affix or non-affix). The goal in dividing the analysis according to these features is to allow for the coverage of all in-scope modifier-like possessive constructions, while capturing as many generalizations as possible.

### 4.3 *Specifier-like possessive constructions*

The goal of this section is to outline an analysis for specifier-like possessive constructions that captures as many generalizations about these constructions as possible, while being adaptable to the various subtypes of specifier-like constructions, including unmarked constructions, as well as single- and double-marked constructions.

Specifier-like possessives are distinguished from modifier-like possessives by the criteria outlined in section 3.3.3.1 above, including whether the possessor may co-occur with other specifiers (such as determiners) and the ordering of possessors relative to modifiers of the

possessum. The distinction between specifier-like and modifier-like possessors holds in unmarked constructions as well as in marked constructions, with the same tests being used to distinguish the two.

In general, the analysis for specifier-like possessive constructions has the possessor selecting for its possessum via its SPEC list and the possessum selecting for its possessor via its SPR list. In marked constructions, the possessive semantics are carried by one of the two constituents, and they are joined by a head-specifier phrase structure rule; in unmarked constructions, the possessive semantics aren't carried by an otherwise unmarked noun, lest they appear in incorrect contexts. Instead, a specialized binary phrase type is introduced which carries the correct possessive semantics, and the preexisting phrase structure rules (*head specifier* in this case) aren't used.

#### 4.3.1 Unmarked possessive phrases

The analysis for the unmarked specifier-like construction coincides in most points with the analysis for the unmarked modifier-like construction. As in that case, the possessor and possessum are joined together by a specialized phrase structure rule, called *poss-phrase*, which performs the necessary work of adding possessive semantics. The primary difference between the specifier-like and modifier-like constructions is that the possessor satisfies the possessum's need for a specifier in the specifier-like construction, while it does not in the modifier-like construction. For that reason, the parent of the specifier-like *poss-phrase* has the constraint [ SPR ⟨ ⟩ ], rather than [ SPR ⟨ X ⟩ ]. The *poss-phrase* also contributes the quantifier relation for the possessum. The specifier-like possessive phrase rule is shown in (63) below.

Although the specifier-like rule for unmarked constructions could likely be modeled as a subtype of the *head-spec* rule in the Grammar Matrix, priority is given to capturing the generalizations between the unmarked specifier-like and modifier-like constructions. These constructions share unique semantic contributions, but not syntactic constraints, such as those found on the *head-specifier* rule. For this reason, the *poss-phrase* rule does not inherit



the possessum is constrained to be [ COG-ST *uniq-id* ], as it would be when accompanied by a definite article such as the English *the*. The feature COG-ST (from ‘cognitive status’) refers to the definiteness of the sign it appears on Borthen and Haugereid (2005). Should the assumption that specifier-like possessors always contribute a definite quantifier fail to hold cross-linguistically, it would be appropriate to replace this value with a more general type that subsumes *uniq-id*, such as *uniq-or-less*. For a full discussion of the various values of COG-ST and their relationships to one another, see Borthen and Haugereid.

#### 4.3.2 *Marked possessive phrases*

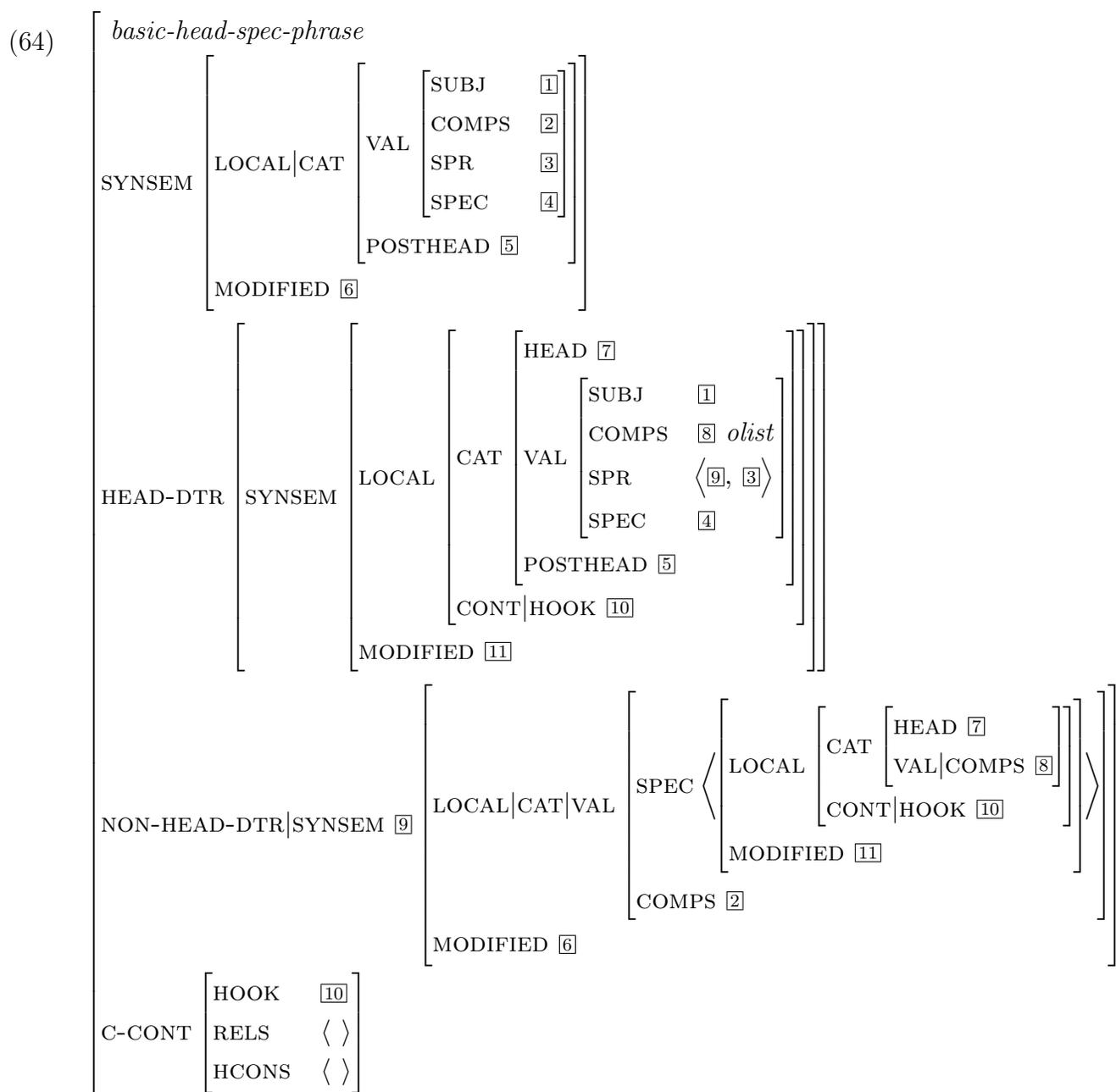
In this section, I outline the analysis for specifier-like possessors where possessive marking of some kind appears in the possessive construction. I begin by discussing the elements of this analysis that are shared by all such constructions. Then I discuss particular variations to this analysis made when the marking is carried by the possessor (section 4.3.2.1), the possessum (section 4.3.2.2), or both possessor and possessum (section 4.3.2.3).

These constructions are all amenable to analysis as instances of head-specifier phrases. No special binary phrase types are added in these cases. The head-specifier rule provided by the Grammar Matrix is shown in (64) below.<sup>10</sup> Its supertypes are *head-valence-phrase*, *phrasal*, and *binary-headed-phrase*.

Though there are advantages to using pre-existing phrase structure rule types, as discussed in section 4.2.2, using a preexisting head-specifier rule to model possession does present two major difficulties which must be resolved: first, the fact that head-specifier rules are non-head-compositional means that to use them unaugmented would lead to incorrect semantic representations; second, cases where the possessive phrase order doesn’t match the order of the preexisting head-specifier rule require additional adaptation. In addition, possessive features are added in order to facilitate the modeling of certain constructions.

---

<sup>10</sup>The rule as shown in (64) actually combines the type definitions of *basic-head-spec-phrase* and its immediate supertype, *basic-head-spec-phrase-super*, since constraints on both are relevant to this discussion.



**Addressing the non-head-compositionality of *head-spec*** A head-specifier rule is instantiated by the word order library in all languages that have determiners (Bender and Flickinger, 2005). This head-specifier rule (called *head-spec*) is non-head-compositional—that is, the HOOK value of the parent of the rule (which contains all the semantic information that is made available to other constituents) is identified with the HOOK value of the non-

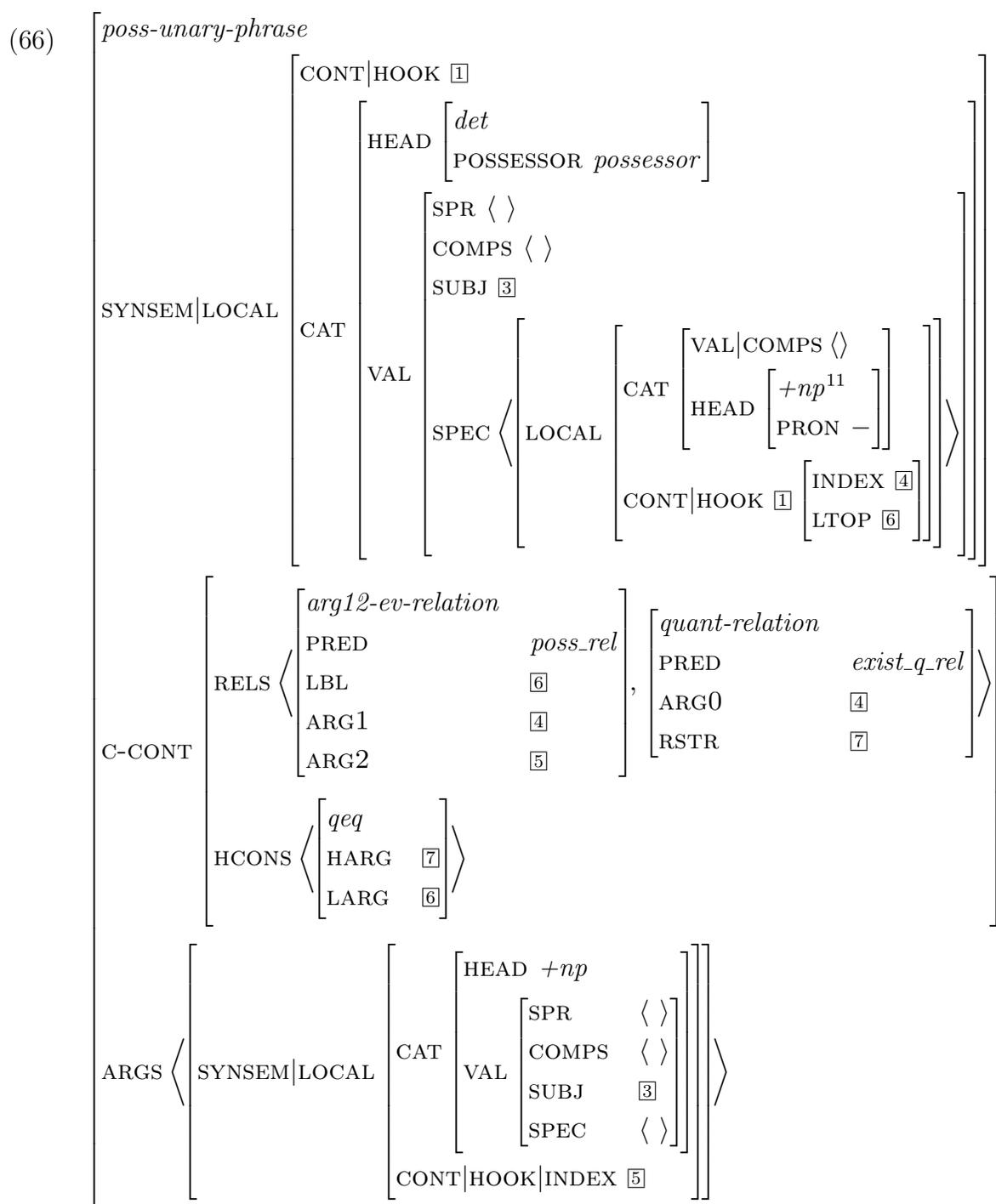
head daughter, unlike most other headed rules, where the HOOK of the parent is identified with the HOOK of the head daughter.

This property makes the *head-specifier* rule insufficient to model possessive phrases by itself. The INDEX value of the head-daughter of a possessive phrase ought to be identified with the INDEX value of the whole phrase—that is, the INDEX value of *the dog's paw* is the same as the INDEX value of *paw*, not of *dog*. Generally, in Grammar Matrix grammars, the *head-spec* phrase is used to combine determiners and nouns. Non-head-compositionality is not problematic in these cases because of the way in which determiners handle their semantics. As seen in the type *basic-determiner-lex* in `matrix.tdl`, determiners identify the INDEX of the thing they specify with their own INDEX:

$$(65) \left[ \begin{array}{l} \textit{basic-determiner-lex} \\ \text{SYNSEM.LOCAL} \left[ \begin{array}{l} \text{CAT|VAL|SPEC|FIRST|LOCAL|CONT|HOOK|INDEX } \boxed{1} \\ \text{CONT|HOOK|INDEX } \boxed{1} \end{array} \right] \end{array} \right]$$

In order to use the *head-spec* rule to model possession, it is tempting to simply have any marking that attaches to the possessor do this same thing—identify the possessum's INDEX with its own. Unfortunately, this doesn't work when the specifier is a noun with its own semantic content. Taking the case of a possessor-marking affix as an example, the lexical rule that added this affix would have to identify the INDEX of the item on the possessor's SPEC list with its own index. However, this would happen before the possessor picked up its own determiner. When it did so, it would present to the determiner the possessum's INDEX value as its own, leading to a broken semantic representation, wherein the possessor's determiner quantified the possessum.

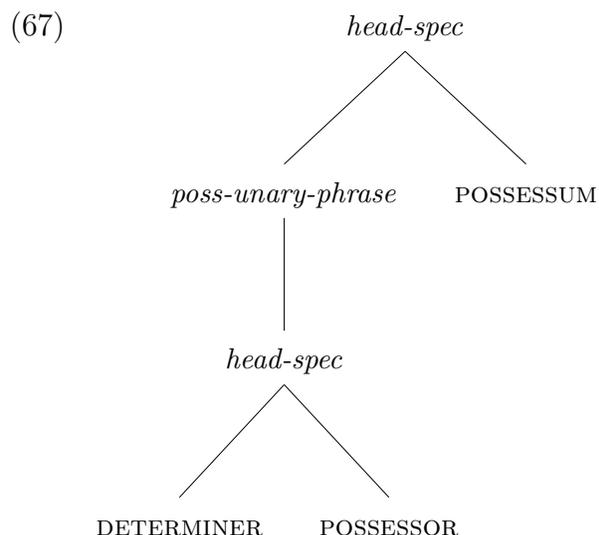
Instead of taking this approach, this problem is solved by introducing a unary phrase rule that takes the possessor along with its determiner as its daughter and does the work of identifying the possessum's INDEX with its own INDEX. This rule also does the work of introducing the *poss\_rel* and other necessary semantic information:



A schematic view of the functioning of *poss-unary-phrase* is given in the abbreviated parse tree in (67), where LEAF NODES represent actual words and *non-leaf nodes* represent

<sup>11</sup>The abbreviation *+np* stands for *noun or adposition* in the Grammar Matrix.

phrase structure rules. As can be seen, the possessor first combines with its determiner, and then this complex passes through *poss-unary-phrase*, before joining with the possessum.



This unary rule includes several important constraints on both its single daughter and its parent. First, the parent of this rule selects for the possessum via its SPEC list, which has one thing on it which is constrained to be [ HEAD *noun* ]. Note that the possessum may not be a pronoun, as is the case in all possessive constructions. Second, the possessor is constrained to be saturated for all VAL features. That is, any possessor must be a full noun phrase, and nominal constituents with unsaturated VAL features cannot take this spot. This means that the possessor will have either joined with a required determiner or have passed through the *bare-np* rule, which allows certain classes of nouns (e.g. pronouns in English) to appear without a determiner, but still be quantified as needed. Finally, the parent has the HEAD type *det*, which allows it to act as the specifier of a *noun-lex*. Further constraints on the daughter of the unary rule are imposed in the case where the possessor is marked, as outlined in section 4.3.2.1 below, in order to ensure that the correct markers appear on all possessors.

This unary rule does the bulk of the work necessary to achieve the target semantic representation for the possessive phrase as well. It introduces the *poss-rel* and the quantifier

for the possessum (along with the appropriate HCONS values), identifying the appropriate arguments of both with the indices of possessor and possessum. These semantic constraints, combined with constraints on the VAL features of both parent and daughter allow this unary rule to appear in the specifier slot of any generic head-specifier rule and produce the correct target semantics.

**Addressing order variation** As mentioned in section 4.2.2 above, there is no guarantee that all languages will have the same relative ordering of possessor and possessum as they have for generic specifiers and heads. For example, in Irish, the possessive phrase is a head-initial *head-spec* phrase, but the *head-spec* phrase consisting of a determiner and a noun is head-final:

(68) rothar Sheain  
 bike John-GEN  
 ‘John’s bike’ [gle] (Lyons, 1986, 140)

(69) an bó  
 the cow  
 ‘the cow’ [gle] (Brothers, 1910, 24)

The approach taken by Zamaraeva and Howell (To be presented) wherein they use the feature INIT to manipulate the ordering of the *head-comps* phrase can be adapted to this scenario. However, a new feature must be introduced,<sup>12</sup> called SPEC-INIT, which is located at HEAD. Specifiers that precede their heads are designated [ SPEC-INIT + ], while those that follow their heads are constrained to be [ SPEC-INIT – ]. In the case of possessors, this constraint is located on the unary rule discussed above. Finally, the variously ordered instantiations of the *head-spec* rule are appropriately constrained for SPEC-INIT on their non-head daughters.

---

<sup>12</sup>The feature INIT cannot be used to account for order variation in both head-complement and head-specifier constructions. A given constituent could conceivably appear in differently ordered head-complement and head-specifier constructions, effectively requiring that it have a positive INIT value in one context and a negative value in another. Since a single constituent cannot have two conflicting values for a single feature, a new feature must be introduced.

**Possessive features** As noted in section 4.2 above, there are two features, POSSESSOR and POSSESSUM, which are used in modeling possessive constructions. These features are employed in all marked constructions, including specifier-like constructions, although they are particularly necessary in modifier-like constructions. See section 4.2.2 above for further discussion on these features.

All marked possessive constructions with specifier-like possessors employ this analysis as recounted to this point, including the use of a unary phrase structure rule, the features POSSESSOR and POSSESSUM, the feature SPEC-INIT, and the case system (in cases of affixal marking, as discussed in section 4.2.2 above). Now I turn to the specific extensions of this analysis in the case of various marking patterns.

#### 4.3.2.1 Possessor-marking

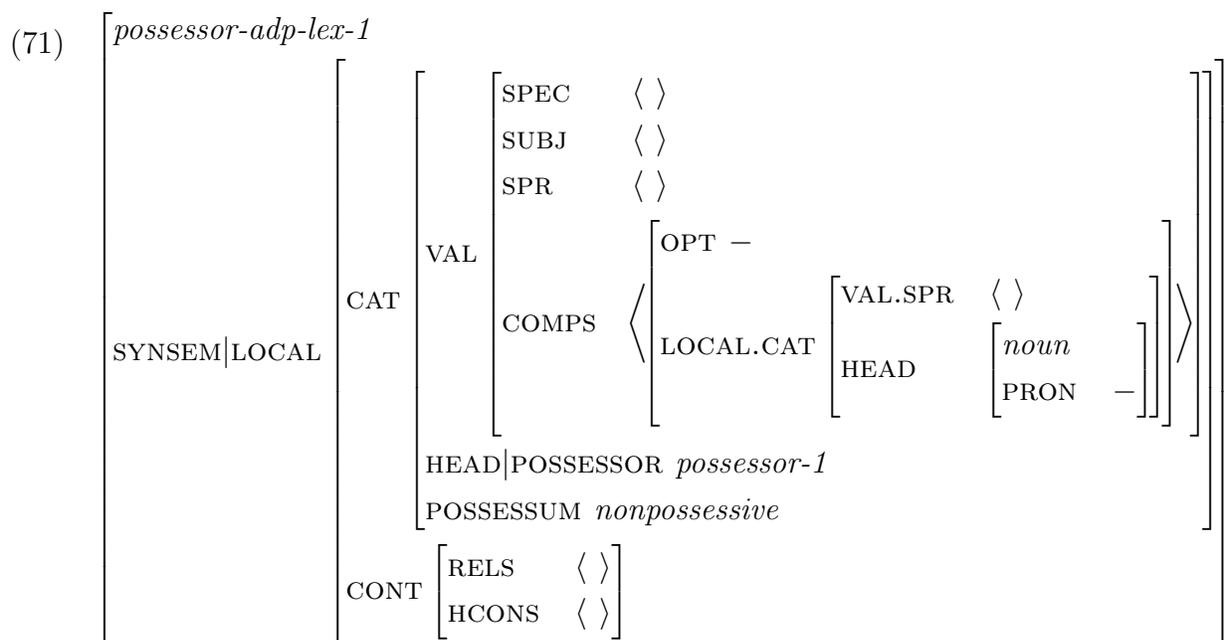
The possessor may be marked by either an affix or a separate word. In either case, the marker on the possessor must carry the feature [ POSSESSOR *possessor* ].

**Affixal marking** The possessor-marking affix is added by a lexical rule (shown in (70) below). Since both the semantics of possession and the selection for the possessum are handled by the unary rule (see (66)), this inflectional rule only adds the affix and sets the value of POSSESSOR to *possessor-n* (where *n* is the appropriate number for a given strategy). The unary rule then constrains its daughter to be [ POSSESSOR *possessor-n* ]. The library designates all other nouns as [ POSSESSOR *nonpossessive* ] so that the unary rule will only accept as its daughter nouns that have been marked as possessors. This prevents unmarked nouns from incorrectly appearing in the spot of the possessor.

$$(70) \left[ \begin{array}{l} \textit{possessor-lex-rule-1} \\ \text{SYNSEM|LOCAL|CAT} \left[ \begin{array}{l} \text{VAL} \quad \boxed{1} \\ \text{HEAD} \quad \left[ \begin{array}{l} \textit{noun} \\ \text{POSSESSOR} \quad \textit{possessor-1} \end{array} \right] \end{array} \right] \\ \text{DTR|SYNSEM|LOCAL|CAT|VAL} \quad \boxed{1} \end{array} \right] \right]$$

**Non-affixal marking** As in the modifier-like case discussed above in section 4.2.2, here

the possessor-marking word is analyzed as an adposition which takes the possessor as its complement. This adposition has one item — the possessum — on its SPEC list, and as above, the possessum is constrained to have saturated SPR value.



#### 4.3.2.2 Possessum-marking

This section gives the analysis for specifier-like constructions in which the possessum is marked. My analysis of these constructions doesn't conform to the generalization that the *poss\_rel* always be borne by the marked constituent (if any). Instead, the *poss\_rel* in this case comes in with the unary rule (shown in (66)). This is necessary because the unary rule that attaches to the possessor identifies its own HOOK value with the HOOK value of the possessum. The HOOK feature is the means by which a constituent exposes its semantics to other elements of a phrase. Since the possessor has taken on the possessum's HOOK value, its own semantics are not exposed and therefore the possessum has no access to its semantics. If the possessum were to carry the *poss\_rel*, it would not be able to access the possessor's actual HOOK and INDEX values in order to correctly identify them with the ARG2 or the *poss\_rel*. Instead, the possessive semantics are handled here by the unary rule (see (66)) that attaches to the possessor, despite the fact that there is no overt marking on the possessor.

**Affixal marking** The possessum-marking rule adds an affix, as well as marking the possessum as [ POSSESSUM *possessum-n* ] (where *n* is the appropriate number for a given strategy). The possessum also requires that its specifier—that is, the possessor—have the feature [ POSSESSOR *possessor-n* ]—that is, that it have passed through the unary phrase structure rule that adds this feature.

$$(72) \left[ \begin{array}{l} \textit{possessum-lex-rule-1} \\ \\ \text{SYNSEM|LOCAL|CAT} \left[ \begin{array}{l} \text{HEAD} \left[ \begin{array}{l} \textit{noun} \\ \text{POSSESSOR } \textit{nonpossessive} \end{array} \right] \\ \text{POSSESSUM } \textit{possessum-1} \\ \text{VAL } \boxed{2} \left[ \text{SPR} \left\langle \left[ \text{LOCAL|CAT} \left[ \begin{array}{l} \text{VAL|SPR } \langle \rangle \\ \text{HEAD|POSSESSOR } \textit{possessor-1} \end{array} \right] \right] \right\rangle \right] \end{array} \right. \\ \\ \text{DTR|SYNSEM|LOCAL|CAT} \left[ \begin{array}{l} \text{POSSESSUM } \textit{nonpossessive} \\ \text{HEAD} \left[ \text{PRON } - \right] \\ \text{VAL } \boxed{2} \end{array} \right] \\ \\ \text{C-CONT} \left[ \begin{array}{l} \text{HCONS} \quad \langle \rangle \\ \text{RELS} \quad \langle \rangle \end{array} \right] \end{array} \right]$$

**Non-affixal marking** A non-affixal marker on a possessum is analyzed as a noun which takes the possessum as its complement, as was the case in the modifier-like scenario. However, where the modifier-like scenario required that the possessum-marking noun attach before any determiners, here the possessum takes no determiner outside of the specifier-like possessor, and so this issue does not arise.



or non-affix). Dividing the analysis according to these features allows for the coverage of all in-scope specifier-like possessive constructions while also capturing possible generalizations over differing scenarios.

#### **4.4 Possessor pronouns**

In addition to strategies to link common nouns together in possessive phrases, many languages also have a class of specialized personal pronouns that act as possessors. In general, a possessor pronoun may take the form of an affix on the possessum, or of an independent word. Both affixal and non-affixal possessor pronouns have modifier-like variants (where the possessum is permitted to still combine with a determiner) and specifier-like variants (where any required determiner on the possessum is replaced by the possessor). Non-affixal possessor pronouns are discussed in section 4.4.1 and affixal pronouns in section 4.4.2.

##### *4.4.1 Possessor pronouns as independent words*

When possessor pronouns are independent words, they may be inflected in the same manner as other nouns are when acting as possessors, or they may be entirely novel forms. In the former case, these pronouns can be accounted for as simply one instance of a given possessive strategy, which applies equally as well to common nouns as to pronouns. Possessor pronouns that are entirely novel forms however, require different treatment. This library allows for them to be entered instead as classes of possessor pronouns in the questionnaire.

Note that the case where a possessor pronoun takes genitive case affixes that are shared by common nouns is parallel to the case of common genitive nouns acting as possessors. All genitive case constructions are analyzed as unmarked constructions where the possessor happens to be constrained to have genitive case (see section 4.2.1). As long as pronouns are permitted by the user to function as possessors in these constructions and the possessor is constrained to be [ CASE *gen* ], genitive case pronouns will act as possessors as expected.

Of the two possible variants, the modifier-like possessor pronoun has a more complex lexical entry, shown in (74):

(74) *poss-pron-1-noun-lex*

SYNSEM	LOCAL	CAT	VAL	$\left[ \begin{array}{l} \text{SPR} \left\langle \left[ \begin{array}{l} \text{LOCAL}   \text{CAT}   \text{HEAD } \textit{det} \end{array} \right] \right\rangle \\ \text{OPT } + \\ \text{COMPS } \textit{olist} \\ \text{SUBJ } \textit{olist} \end{array} \right]$
			HEAD	$\left[ \begin{array}{l} \textit{noun} \\ \text{PRON } + \\ \text{POSSESSOR } \textit{possessor-pron-1} \\ \text{MOD} \left\langle \begin{array}{l} \text{OPT } - \\ \text{LOCAL} \left[ \begin{array}{l} \text{CAT} \left[ \begin{array}{l} \text{HEAD}   \text{PRON } - \\ \text{VAL}   \text{SPR} \langle \boxed{1} \rangle \end{array} \right] \\ \text{CONT.HOOK} \left[ \begin{array}{l} \text{INDEX } \boxed{4} \\ \text{LTOP } \boxed{5} \end{array} \right] \end{array} \right] \end{array} \right\rangle \end{array} \right]$
			HOOK	$\left[ \begin{array}{l} \text{INDEX } \boxed{2} \left[ \begin{array}{l} \text{NUM } \textit{sg} \\ \text{PER } \textit{1st} \end{array} \right] \\ \text{LTOP } \boxed{1} \end{array} \right]$
			RELS	$\left\langle \begin{array}{l} \left[ \begin{array}{l} \textit{arg12-ev-relation} \\ \text{PRED} \quad \textit{poss\_rel} \\ \text{LBL} \quad \boxed{5} \\ \text{ARG1} \quad \boxed{4} \\ \text{ARG2} \quad \boxed{2} \end{array} \right], \\ \left[ \begin{array}{l} \textit{noun-relation} \\ \text{PRED } \textit{pron\_rel} \\ \text{LBL } \boxed{1} \\ \text{ARG0 } \boxed{2} \left[ \begin{array}{l} \text{COG-ST } \textit{activ-or-more} \\ \text{SPECI } + \end{array} \right] \end{array} \right] \end{array} \right\rangle$
			HCONS	$\langle \rangle$
LKEYS ALTKEYREL $\boxed{3}$ <sup>13</sup>				

<sup>13</sup>Note that the (*pron\_rel*) is identified with the ALTKEYREL value in this lexical entry. This is in order

The modifier-like possessor pronoun must carry all necessary semantic and VAL features itself. By contrast, a specifier-like pronoun acts as the daughter of the unary rule which adds all relevant possessive semantics, as well as adding the correct VAL features (see (66) above for details on the unary rule and (67 for a tree-based view of its functioning)). This means that a specifier-like pronoun is essentially a personal pronoun with the correct POSSESSOR feature value in order to act as a daughter of the unary rule.

Note that the possessor pronoun shown in (74) is [ SPR < [ OPT + ] > ]. Here, the feature OPT (from ‘optional’) is used to indicate not simply that a specifier is optional (as this feature was originally meant to convey), but that the specifier *must* be dropped. This constraint prevents the possessor pronoun from picking up a determiner and instead makes it pass through the *bare-np* rule, which contributes the necessary quantifier for the pronoun.

No possessor pronoun construction is modeled with a head-complement phrase. As discussed in section 4.2.2.2, the head-complement phrase is only strictly necessary in those cases where the head is marked and the possessor is unmarked. Since the possessor pronoun is, by definition, inherently marked as a possessor, this situation never arises, and head-modifier phrases (such as the *head-adj-int* rule) can be used.

#### 4.4.2 Possessor pronouns as affixes

Possessor pronouns can also take the form of affixes on the possessum. In this case, the inflected possessum (plus any necessary determiners) constitutes the entire possessive phrase. As is the case in all other scenarios, the user is given the option of allowing the pronoun inflection to either satisfy or not satisfy the possessum’s requirement for a determiner — that is, something analogous to a specifier-like and a modifier-like analysis is still possible in this case. The modifier-like scenario is seen in the Hungarian example in (32b), reproduced here, while the specifier-like scenario is seen in Finnish:

---

to allow the person, number, and gender features of the pronoun (which are shown here at the INDEX (marked as  $\boxplus$ ) to be identified easily with the ARG0 of the *pron\_rel* easily. Manipulating the constraints of an item on the RELS list directly is difficult, and so identifying this relation with the ALTKEYREL helps make it easier to access.

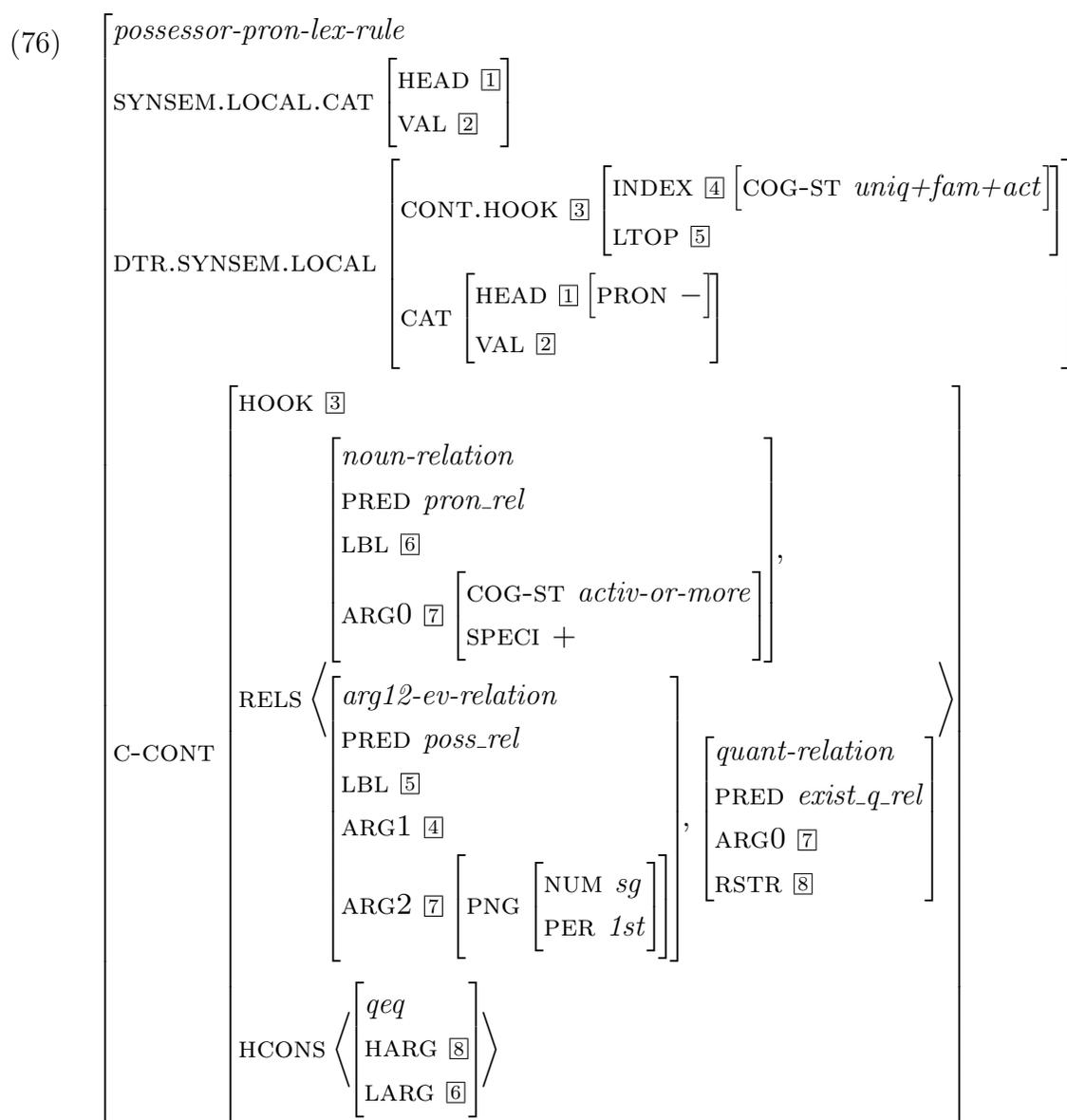
- (75) a. Hungarian:  
       a kalap-unk  
       the hat-POSS.1PL  
       ‘our hats’ [hun] (Szabolcsi, 1994, 8)
- b. Finnish:  
       velje-ni  
       brother-POSS.1SG  
       ‘my brother’ [fin] (Karlsson, 2015, 125)

These possessor pronoun affixes are instantiated as inflectional rules in the grammar. An example of an inflectional rule that adds a first-person singular possessor pronoun affix is shown in (76). This is the ‘modifier-like’ variant of the possessor pronoun affix — that is, it does not assume that the possessum’s SPR value can be satisfied by the addition of the pronoun affix. Rather, it simply copies up the VAL features of the possessum intact.

The ‘specifier-like’ variant of the possessor pronoun affix functions similarly, but it modifies the VAL features of the possessum, satisfying its need for a specifier. A noun with a specifier-like possessor pronoun affix should not combine with a determiner; however, it should presumably be able to be modified by adjectives, which only modify nouns with non-empty SPR lists. For this reason, the best solution is to have the specifier-like possessor pronoun affix change the possessum’s specifier to have the value [ OPT + ], which will ensure that it cannot combine with a determiner, even though it still has the property [ SPR < X > ],<sup>14</sup> making it amenable to modification by adjectives. This analysis was not fully implemented in the library prior to evaluation. Instead of marking the possessum’s specifier [ OPT + ], the possessor pronoun affix simply saturated the possessum’s SPR value, making it impossible for adjectives to attach. Though this error was not discovered in evaluation, it was fixed in post-evaluation editing (see section 6.4).

---

<sup>14</sup>This is due to enforced constraints on the head-specifier rule in the Grammar Matrix, which requires that the constraint [ OPT – ] be present on the SPR value.



#### 4.4.3 Summary

Possessor pronouns appear both as independent words, and as affixes on the possessum. In either case, this analysis allows them to be modeled as either modifier-like or specifier-like, in keeping with the behavior of a given language.

## 4.5 Agreement

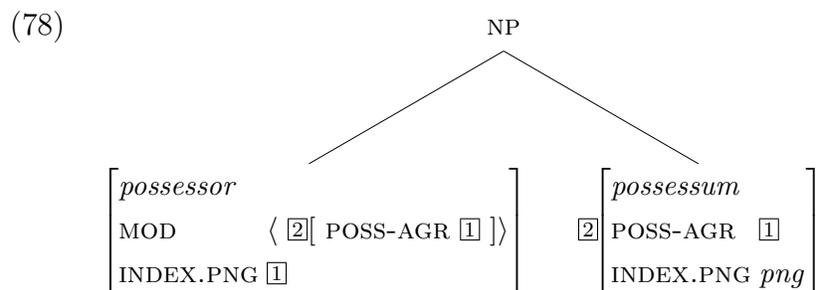
Many languages require that the possessum agree with the possessor (or vice versa) in semantic features such as person, number, or gender. Some languages also require agreement in syntactic features, including case. This analysis accounts for agreement in either direction between possessor and possessum for all semantic features defined in the Grammar Matrix, including person, number, and gender. No adaptation would be required to also cover any other semantic features (e.g. animacy) that the user might define. However, agreement in features such as case that are syntactic rather than semantic (i.e. which reside at HEAD and not HOOK), cannot simply use the same analysis. Case agreement is handled only for one scenario—modifier-like possessor pronouns. Other instances of case agreement are left as potential avenues for future work.

### 4.5.1 Semantic feature agreement

Accounting for agreement between possessor and possessum requires that, in some circumstances, a second set of agreement features be added to a single constituent, in addition to its inherent semantic features. To see why this is necessary, first consider a simple case where a single set of features suffices. When the possessor is specifier-like and the possessum agrees with it, it would not strictly be necessary to have any additional agreement features encoded on any one constituent. The possessum can simply constrain the relevant person, number, and gender features of the possessor, which appears on its SPR list. Since the head and non-head daughters both select for each other in the head-specifier schema, this analysis works equally well in the case where the possessor agrees with the possessum. However, when the possessor is modifier-like, possessor and possessum are joined by a head-modifier or head-complement rule which, as noted before, has no such mutual selection. The possessor can constrain the features of the possessum, which appears on its MOD list, but the possessum has no access to its possessor's features. This pattern is seen in the following example from Hungarian, where the possessum agrees with a modifier-like possessor:

- (77) az én kalap-ja-i-m  
 the I hat-POSS-PL-1SG  
 ‘my hats’ [hun] Laczko (2007)

Since the possessum cannot select its modifier, instead the possessum must somehow ‘publish’ the person, number or gender features it agrees with, so that the possessor can select for a possessum with the correct agreement features. This means it is necessary for the possessum to carry two sets of semantic features: the inherent person, number, and gender features it has as a noun; and the person, number, and gender features that it agrees with. The former are found (as usual) at INDEX.PNG, while the latter are in the new head feature I introduce, called POSS-AGR, which is located in the POSSESSUM feature in this example (though in the case of agreement in the opposite direction, POSS-AGR is located in the POSSESSOR feature). The possessum can then do the work of identifying the possessor’s agreement features with its own features, as sketched in the tree in (78):



Since this POSS-AGR feature is necessary in some analyses, it is used as the default means of modeling agreement in all scenarios where it appears (though the POSS-AGR feature is not added when no agreement exists).<sup>15</sup> Note, however, that the particulars of the above analysis given for the Hungarian scenario are exceptional in one regard. In it, the *agreed with* element does the work of identifying the other element’s POSS-AGR features with its own INDEX.PNG features. This is necessary because of the way in which selection in head-modifier rules works. In all other scenarios, the *agreeing* constituent does all of the work: as before, it retains its own semantic features, and then adds a second set of agreement features at POSS-AGR (under POSSESSOR if the constituent is a possessor, and under POSSESSUM if it

is a possessum). Then the agreeing constituent identifies these POSS-AGR features with the INDEX.PNG of the agreed-with constituent. The effect is the same whether this identification is encoded on the agreeing or the agreed with element; it is simply the case that in one scenario, illustrated by Hungarian above, only one element is capable of doing this work.

#### 4.5.2 Case agreement

In the scenario where the possessor is a common noun, rather than a pronoun, case agreement in possessive phrases is comparatively rare, primarily being attested in the languages of Australia and India, and in a few other languages, such as Old Georgian (Grashchenkov, 2005, 39). However, case agreement of the possessor with the possessum is well attested in possessor pronouns, as in this example from Russian:

(79) Russian:

moj                ded                byl                okhotnik-om  
1SG.POSS.NOM grandfather.NOM be.SG.MASC.PST hunter-INST

‘My grandfather was a hunter’ [rus]

(Russian National Corpus, 2018)

oni s    mater’-ju    po-shli            k moj-emu    ded-u  
3PL with mother-INST PFV-go.PL.PST to 1SG.POSS-DAT grandfather-DAT

‘They and mother went to my grandfather’ [rus] (Russian National Corpus, 2018)

Case agreement is more complicated than semantic feature agreement because CASE is a syntactic feature, which cannot be stored under POSS-AGR. In the current implementation, POSS-AGR takes a value of type *png*, which only contains semantic features. This allows for the PNG and POSS-AGR features to be identified with one another directly in order to model agreement. Fortunately, case agreement is always in just a single direction—the possessor agreeing with the possessum, not vice versa. This means that the issues surrounding Hungarian, discussed above, will not apply here, since the scenario with agreement of a possessum

---

<sup>15</sup>In the interest of generating more elegant grammars, it would be preferable to introduce the POSS-AGR only in those scenarios where it is strictly required. Changing the library in this way is left as future work.

with its possessor drove the analysis in that case. It should be sufficient to simply have the possessor constrain the CASE value of the thing on its MOD or SPEC list, as appropriate.

Currently, this is implemented for only one scenario: modifier-like possessor pronouns. This can be done rather simply in this case, whereas the specifier-like case is more complicated, because the possessor pronoun cannot constrain the features of the possessum directly, but must do so via the unary rule that is present in these analyses. Since the possessor pronoun's own CASE value is already marked as *poss-case* in these scenarios, there is no obvious solution to this issue. However, implementing this analysis for modifier-like possessor pronouns allows one to model many if not most instances of case agreement in the possessor. Modeling case agreement on specifier-like possessor pronouns and on all common noun possessors is left as future work.

#### 4.5.3 *Summary*

Agreement of semantic features between of possessum with possessor — and vice versa — is attested in many languages, and is modeled by the introduction of a new location (POSS-AGR) in the feature matrix for an existing set of semantic features. Case agreement of possessor with possessum is also attested in a limited set of languages, and a limited subset of this phenomenon is modeled: modifier-like possessor pronouns.

### 4.6 *Summary*

The goal of devising this analysis has been to cover all in-scope phenomena by factoring the typological space according to a limited set of features. This allows for an efficient implementation of the library and captures possible generalizations about the way (for example) that modifier-like constructions or unmarked constructions behave. The phenomena covered include specifier-like and modifier-like possessive constructions; unmarked possessive constructions and cases where the marking takes the form of an affix or an independent word; and marking of the possessor, the possessum, or both. This chapter also includes analyses of possessor pronouns and agreement between possessor and possessum.

## Chapter 5

### IMPLEMENTATION

I implement the analysis set out in chapter 4 above in the Grammar Matrix customization system (Bender and Flickinger, 2005; Drellishak, 2009; Bender et al., 2010) by adding to the customization questionnaire and the grammar-producing Python code. The questionnaire serves to elicit the necessary information from the user-linguist, which is then passed to the grammar-producing Python code and used to generate a grammar. In this chapter, I describe the additions that were made to the questionnaire and the architecture of the grammar-producing code that handles adnominal possession. I also include an example of a choices file that is produced by the questionnaire and the grammar that it generates.

#### **5.1 Questionnaire**

The customization questionnaire is divided into subpages, with one linguistic phenomenon being described on each subpage. Implementing a library for adnominal possession meant entails adding another subpage to the questionnaire. This subpage is divided into two primary sections: one for entering strategies where the possessor is a common noun and one for entering classes of possessor pronouns. The user-linguist can enter indefinitely many strategies or classes of pronouns using the iterator object developed by Drellishak (2009). Many languages have at minimum one strategy for possession between common nouns and one class of possessor pronouns; some languages have many more than this, so a means for entering multiple instances of both is necessary.

The information that is needed in order to construct a given adnominal possessive strategy or pronoun class varies. For example, in most cases it is important to know the form and location of a possessive marker, but these questions are clearly irrelevant in the case where no

marking exists. Also, entry fields for possessor and possessum markers must be kept distinct (since they may co-occur in a single strategy), but the entry fields for possessor markers are irrelevant if the strategy marks only the possessum. For these reasons, I design the questionnaire to adapt to the user's answers, presenting only relevant questions to the user. In the questionnaire itself, this is achieved by toggling the visibility of questions depending on the answers to earlier questions. Here, the same contingency between questions is shown as an indented list below. In general, a question is only asked of the user if it is either (1) at the left-most level of indentation in the list below, or (2) listed directly under an answer supplied by the user to a previous question. As with all other libraries, the answers to these questions are stored in a choices file, which is then parsed by the grammar-producing code.

**Possessive strategies for common nouns:**

- (1) What is the order of constituents in the possessive phrase?
- (2) Is the possessor modifier-like or specifier-like?
- (3) Does a marking appear on the possessor, possessum, both, or neither?
  - (I) IF UNMARKED:
    - (A) Can the possessor in these phrases be an unmarked pronoun?
    - (B) Are there any constraints (e.g. case) on the possessor?
  - (II) IF POSSESSOR-MARKING OR BOTH-MARKING:
    - (A) Is the marker on the possessor an affix or an independent word?
      - (i) IF AFFIX:
        - (a) Does the possessor affix agree with the possessum?
 

**At this point, the user is directed to use the morphology sub-page to enter the actual affix**
      - (i) IF INDEPENDENT WORD:
        - (a) Can the possessor marked by this word be a pronoun?
        - (b) Are there any constraints (e.g. case) on the possessor?
        - (c) Does the possessor-marking word agree with the possessum?

(1) IF NO: **the user is asked to enter the orthographic form of**

**the single possessor-marking word**

- (2) IF YES: **the user is asked to enter all possessor-marking words and their agreement features.**

(III) IF POSSESSUM-MARKING OR BOTH-MARKING:

(A) IF POSSESSUM-MARKING ONLY:

- (i) Are there any constraints (e.g. case) on the possessor?

(B) ELSE:

- (i) Is the marker on the possessum an affix or an independent word?

(a) IF AFFIX:

- (1) Does the possessum affix agree with the possessor?

**At this point, the user is directed to use the morphology subpage to enter the actual affix**

(a) IF INDEPENDENT WORD:

- (1) Does the possessor-marking word agree with the possessum?

(I) IF NO: **the user is asked to enter the orthographic form of the single possessum-marking word**

(II) IF YES: **the user is asked to enter all possessum-marking words and their agreement features.**

**Possessor pronouns:**

- (1) Are these pronouns independent words, or are they affixes?

(I) IF AFFIXES:

(A) Do these affixes agree with the possessum?

(B) If determiners exist, does the possessum noun still appear with a determiner when marked with these affixes?

**At this point, the user is directed to use the morphology subpage to enter the actual affix**

(II) IF INDEPENDENT WORDS:

(A) Do the possessor pronouns occur to the left or the right of the possessum?

- (B) Are these possessor pronouns modifier-like or specifier-like?
- (C) Do these pronouns agree with the possessum?

**At this point, the user is asked to enter the orthographic forms of the possessor pronouns and any agreement features**

- (D) Does the possessum take any marking when it appears with these possessor pronouns?
  - (i) IF YES: **answer questions beginning at III in the common noun questionnaire above.**

Though much of the questionnaire is self-explanatory, several points require clarification. First, the user is not asked to specify feature constraints on every constituent. In particular, the user is not asked to give constraints that apply to the possessum. This is because it is assumed that, as the head noun of the noun phrase, features such as case will be constrained by other parts of the grammar and should not be altered by the adnominal possession library.<sup>1</sup> A possessor, however, is not constrained by anything outside of the adnominal possession library and should be dealt with here. Likewise, constituents that are marked by affixes do not have the option of adding constraints on this page. This is because the actual affixes are added on the morphology subpage (O'Hara, 2008; Goodman, 2013) rather than the adnominal possession subpage, as described in section 2.3. When the user indicates that a strategy includes an affix, a feature corresponding to that strategy or pronoun is created on the morphology page. The user then creates an inflectional rule on the morphology page and gives it the appropriate feature value to link it to the possessive strategy. Constituents marked with affixes can use the analysis already implemented on the morphology page to handle interactions between inflectional affixes, including case, number, gender, etc., and so no other means of adding the correct feature constraints is needed.

Lastly, at several points in the questionnaire, the user is asked to indicate whether a certain constituent may be a pronoun. This is never asked of the possessum, since possessing

---

<sup>1</sup>The fact that this option is missing does make inalienable possession impossible to model in some cases (see section 6.2.4 below), but inalienable possession is not within the primary scope of this thesis.

a pronoun is prohibited in all languages. The user is asked to indicate if the possessor can be a pronoun in cases where the possessor does not take an affixal marking (for the reasons indicated above—the morphology subpage already allows the user to prohibit or allow pronouns to take any affix desired).

## 5.2 Grammar-producing code

This section describes the architecture of the Python code that actually produces the grammar in TDL according to the specifications of the user-linguist. The files containing this code are located within the Grammar Matrix repository<sup>2</sup> at the path `trunk/gmcs/linglib`. This section is organized by filename. The majority of the grammar-producing code for this library was added in `adnominal_possession.py`, but many other files were edited as well.

### 5.2.1 *adnominal\_possession.py*

The file `adnominal_possession.py` contains the bulk of the code necessary to produce the phrase rules, lexical rules, and lexical entries described in the analysis outlined in chapter 4. This section details the logic flow within `adnominal_possession.py`.

The primary function within this file, `customize_adnominal_possession()`, calls three functions: `customize_poss_addenda()`, `customize_np_possession()`, and `customize_pronominal_possession()`. The function `customize_poss_addenda()` does the work of adding the `POSSESSOR`, `POSSESSUM`, and `POSS-AGR` features at the appropriate paths. The other two functions call the rest of the necessary code for adding possessive strategies where the possessor is a common noun (`customize_np_possession()`) and for adding classes of possessor pronouns (`customize_pronominal_possession()`).

Both `customize_np_possession()` and `customize_pronominal_possession()` function in largely the same way in their respective domains: each takes in the choices about the

---

<sup>2</sup>The Grammar Matrix can be found in a Subversion repository at `svn://lemur.ling.washington.edu/shared/matrix/trunk`. The evaluation described in this chapter was carried out on revision number 41612.

ordering of constituents in the possessive phrase, the location of markers, the status of the possessor (modifier vs. specifier) and calls a subset of the following functions, as appropriate, to add phrase rules, inflectional rules, and lexical entries:

1. `customize_poss_rules()`: this function adds any necessary phrase rules to the grammar. In specifier-like strategies, it adds at minimum the *poss-unary-rule*. In many cases, the appropriate binary phrase type already exists (*head-spec*, *head-adj-int*, *head-comp*), but in some cases it does not and must be added, or a variant with the desired ordering must be added. The features `POSTHEAD`, `INIT`, and `SPEC-INIT` are added in this function as well to constrict the order of possessive phrases.
2. `customize_possessor_irules()`, `customize_possessum_irules()`, `customize_possessor_pron_irules()`: These functions add inflectional rules that add affixal markings to the possessor or the possessum, as indicated by the user. Within each of these functions, the logic flow is as follows: a maximally general inflectional rule is added to the grammar; then, the strategy contained in the choices file is parsed to extract such information as the status of the possessor (modifier vs. specifier) and the location of the marking. Based on these, the general inflectional rule is augmented as appropriate with the correct `VAL`, `C-CONT`, `POSSESSOR`, and `POSSESSUM` features. Then, in cases where the user has indicated that there is agreement between possessor and possessum, these functions add a constraint to the inflectional rule that identifies their agreement features (located at `POSS-AGR`) with the features of the agreed-with constituent. Then, these functions call `customize_feature_values()`, defined in `features.py`, which adds the correct agreement features at `POSS-AGR`. Note that since the `morphotactics` library already contains mechanisms for ensuring the correct ordering of affixes, all of these inflectional rules inherit from types that are created in `morphotactics.py`. These rules serve to manipulate the values of the `INFLECTED` feature and ensure that the possessive affixes have the correct morphotactic behavior, as specified by the user. However, the actual inflectional rules that add the appropriate features and phonological material for

possessive constructions are added in `adnominal_possession.py`.

3. `customize_possessor_lexicon()`, `customize_possessum_lexicon()`, `customize_possessor_pron_lexicon()`: These functions do largely the same thing that the inflectional rule-adding functions do, but without any interaction with `morphotactics.py`: they add the correct possessor- or possessum-marking words with the VAL, C-CONT, POSSESSOR, and POSSESSUM features appropriate given the user choices. They also instantiate these lexical items within the `lexicon.tdl` file.

### 5.2.2 *morphotactics.py*

In order to ensure that possessive affix-adding rules are added with all and only the correct supertypes, I made several edits to `morphotactics.py`. First, as a default, the rule type *add-only-no-ccont* is added as a rule supertype for all inflecting rules. This rule requires that its mother and daughter share all syntactic features, and prevents any semantic information from being added via the feature C-CONT. Since many possessive rules have a non-empty C-CONT value—namely, the *poss\_rel* and other necessary parts of the semantics of possessive phrases—I altered `morphotactics.py` to ensure that this supertype not added to possessive lexical rules.

Second, the possessive rules added in `adnominal_possession.py` need to interact correctly with the morphotactics-handling lexical rules that are already produced by the code in `morphotactics.py` which manipulate the value of the INFLECTED feature. This feature handles morphotactic issues such as affix ordering and cooccurrence by keeping track of which inflectional rules have been applied to a lexical entry, which can be applied, and which must be applied. However, these rules are not automatically linked in any way to the inflectional rules added in `adnominal_possession.py`. The inflectional rules that will be employed by the grammar must combine the possession-specific feature constraints from the rules added in `adnominal_possession.py` and the morphotactics-handling constraints on the rules added by `morphotactics.py`.

In order to ensure that the rules added in `adnominal_possession.py` are linked with the morphotactics-handling rules added by `morphotactics.py`, I employ a solution used in the evidentiality library (Haeger, 2017) among others. In addition to the adding one morphotactics-handling rule per position class, `morphotactics.py` adds one lexical rule per lexical rule type defined at a given position class. These automatically added lexical rules already inherit from the morphotactics-handling rule, and so their `INFLECTED` values are handled correctly. In order to link the possessive rules added in `adnominal_possession.py` with the morphotactics-handling rules, the possessive rules are added as a supertype of each lexical rule type generated by `morphotactics.py`. This ensures that each lexical rule type inherits from both the morphotactics handling rule and the possessive rule as defined in `adnominal_possession.py`, giving it the correct possessive properties as well as the desired morphotactic behavior.

### 5.2.3 *lexbase.py*

In this file, the `LexicalRuleType` object which is used within `morphotactics.py` is augmented with the properties, `possessive` and `poss_strat_num`. These fields store information about whether a possessive rule is defined in a given lexical rule type.

### 5.2.4 *features.py*

In this file, the function which normally adds semantic and syntactic features to lexical entries and rules is augmented in order to add these features at the correct paths. While most lexical items only need to have semantic features added at their own `CONT|HOOK|INDEX` path, possessive rules and lexical items may need semantic features to be added at other paths, such as `POSSESSOR.POSS-AGR` or `POSSESSUM.POSS-AGR`. I altered `features.py` in order to ensure that these semantic features are added at the correct paths in the case of possessive entities.

### 5.2.5 *case.py*

Within this file, the intermediate value for CASE, *real-case* is added to the case hierarchy when a possessive strategy is defined and the language has a user-defined case system. As described in section 4.3.2 above, this can be used to stop the possessor from appearing in contexts where it should not appear.

### 5.2.6 *lexical\_items.py*

Within this file, the constraint [ CASE *real-case* ] is added to arguments of verbs, adpositions, and adjectives in cases where a possessive strategy has been defined and the language has case. The feature PRON is also added in this file, which allows the user to designate certain words as pronouns, and therefore not candidate for certain roles within the possessive phrase.

### 5.2.7 *coordination.py*

The phrase rules that build coordinated phrases must be updated to deal with possessive features. Edits are necessary in `coordination.py` in order to ensure that the values of POSSESSOR and POSSESSUM are shared between conjuncts and the parent of a coordinated phrase. Adding these constraints is handled in a function separate from the rest of the coordination-adding code, which is called only if possessive strategies are implemented.

## 5.3 *Example*

To demonstrate the functioning of the questionnaire and the grammar-producing code, in this section I demonstrate how a simple choices file for a pseudo-language is parsed by the grammar-producing code to generate TDL which implements possessive constructions. The pseudo-language is described by the (somewhat abbreviated) choices file shown in figure 5.1. This choices file is output by the completed questionnaire and is used as input to the grammar-producing code. As can be seen in the `adnom-poss` section, this language has a single type of possessive construction (`poss-strat1`), which has possessum-final or-

dering (`poss-strat1_order=head-final`). The possessor is specifier-like (`poss-strat1_mod-spec=spec`) and there is overt marking on the possessor (`poss-strat1_mark-loc=possessor`). The marking takes the form of a separate word or clitic (`poss-strat1_possessor-type=non-affix`), which does not agree with the possessum (`poss-strat1_possessor-non-affix-agr=non-agree`) and precedes the word it marks (`poss-strat1_possessor-mark-order=head-initial`). Since this is a pseudo-language, the possessor-marking word is spelled *POSSESSOR* and other lexical items are similarly named, e.g. *noun1*, *noun2*.

Upon receiving this choices file, the grammar-producing code contained in `adnominal_possession.py` fires two primary functions: `customize_poss_rules()`, which handles adding and augmenting phrase structure rules, and `customize_possessor_lexicon()`, which adds possessor-marking words. The first of these functions checks the values for the choices `poss-strat1_mod-spec` and `poss-strat1_order` in order to determine which phrase structure rule is needed to model the possessor's attachment to the possessum (in this case, *head-spec*) and what ordering it should have (*head-final*). In this pseudo-language, a *head-spec* rule is already instantiated with the correct order, since the choices file indicates that it has determiners which precede their head nouns. Thus, `customize_poss_rules()` does not add a *head-spec* rule. It is also the job of `customize_poss_rules()` to determine whether or not any phrase structure rules need to be added in order to model the attachment of the possessor-marking word to the possessor. In this case, the default order of *head-comp* is *head-initial*, and the desired ordering of possessor-marker and possessor is likewise *head-initial*. Thus, no phrase structure rules are added by `customize_poss_rules()` at any point.

The other function that fires is `customize_possessor_lexicon()`. This function is triggered solely on the appearance of the choice `poss-strat1_possessor-type=non-affix`, and so in all cases, it adds a maximally general type definition for a possessor-marking adposition that is appropriate for all such constructions. For example, the maximally general type definition includes such constraints as the *poss-rel* and the requirement that the marker take a complement, which is the possessor. The function then further parses the choices

```

version=28

section=word-order
word-order=svo
has-dets=yes
noun-det-order=det-noun
has-aux=no

section=number
  number1_name=sg
  number2_name=pl

section=person
person=1-2-3
first-person=none

section=case
case-marking=none

section=adnom-poss
  poss-strat1_order=head-final
  poss-strat1_mod-spec=spec
  poss-strat1_mark-loc=possessor
  poss-strat1_possessor-type=non-affix
  poss-strat1_possessor-mark-order=head-initial
  poss-strat1_possessor-agr=non-agree
  poss-strat1_possessor-orth=POSSESSOR

section=lexicon
  noun1_name=common
  noun1_feat1_name=person
  noun1_feat1_value=3rd
  noun1_det=obl
  noun1_stem1_orth=noun1
  noun1_stem1_pred=_noun1_n_rel
  noun1_stem2_orth=noun2
  noun1_stem2_pred=_noun2_n_rel

section=morphology
noun-pc1_name=num
noun-pc1_obligatory=on
noun-pc1_order=suffix
noun-pc1_inputs=noun1
  noun-pc1_lrt1_name=singular
  noun-pc1_lrt1_feat1_name=number
  noun-pc1_lrt1_feat1_value=sg
  noun-pc1_lrt1_feat1_head=itself
  noun-pc1_lrt1_lri1_inflecting=no
  noun-pc1_lrt2_name=plural
  noun-pc1_lrt2_feat1_name=number
  noun-pc1_lrt2_feat1_value=pl
  noun-pc1_lrt2_feat1_head=itself
  noun-pc1_lrt2_lri1_inflecting=yes
  noun-pc1_lrt2_lri1_orth=PL

```

Figure 5.1: The choices file for a pseudo-language with one possessive strategy, wherein the specifier-like possessor is affixally marked.

file in order to add appropriate refinements to this general type definition. In this case, the choice `poss-strat1_mod-spec=spec` tells the function to add a non-empty SPEC list corresponding to the possessum to the general type definition. In the end, the lexical entry for the possessor-marking adposition is as the TDL shown in figure 5.2.

```

possessor-adp-lex-1 := two-rel-adposition-lex &
  [ SYNSEM.LOCAL [ CAT [ VAL [ SPEC < >,
                           SUBJ < >,
                           SPR < >,
                           COMPS.FIRST [ OPT -,
                                          LOCAL.CAT [ VAL.SPR < >,
                                                       HEAD noun &
                                                       [ PRON - ] ] ] ],
                           HEAD.POSSESSOR possessor-1,
                           POSSESSUM nonpossessive ],
    CONT [ ICONS <! !>,
          RELS <! !>,
          HCONS <! !> ] ] ].

```

Figure 5.2: TDL definition of a possessor-marking adposition

In addition to the TDL added by the functions `customize_poss_rules()` and `customize_possessor_lexicon()`, there are some things added to the grammar in all cases where a possessive strategy is defined, regardless of its nature. These include a set of possessive features, as well as extra constraints to preexisting phrase structure rules, as show in figure 5.3.

#### 5.4 Summary

In this chapter, I have described the implementation of a system to automatically generate a machine-readable HPSG analysis of adnominal possessive constructions. This implementation takes the form of a questionnaire that elicits user input about a language's possessive constructions, and Python-based code that generates the grammar itself, as described above.

```

possession := *top*.
possessive := poss. ; supertype for possessive features
nonpossessive := poss. ; type for nonpossessive
possessor := possessive.
possessum := possessive.
possessive-1 := possessive. ; supertype for strategy 1
possessor-1 := possessor & possessive-1.
possessum-1 := possessum & possessive-1.

basic-bare-np-phrase := [ SYNSEM.LOCAL.CAT [ VAL.SPEC < >,
                                HEAD #head,
                                POSSESSUM #possessum ],
    HEAD-DTR.SYNSEM.LOCAL.CAT [ HEAD #head,
                                POSSESSUM #possessum & nonpossessive ] ].

```

Figure 5.3: TDL definitions of boilerplate additions to the grammar for possessive constructions

## Chapter 6

### EVALUATION

In order to determine how this analysis and implementation cover the described typological space, testing is conducted in three stages. First, the library is tested on a series of ‘pseudo-languages’ — minimally complex invented languages that test different combinations of choices. Second, it is tested on five illustrative languages, which are real languages whose features have served as motivation for the analysis and implementation. Throughout these first two steps, the implementation is still being refined and bugs are being identified and fixed. For the third stage of testing, the implementation is held constant, and a set of five languages is picked at random. The performance of the library on these held-out languages is used to evaluate its completeness, correctness, and generalizability.

In each case, a test consists of a testsuite file and a choices file. The testsuite consists of a list of strings, some of which are grammatical sentences, and some of which are ungrammatical. This allows for the performance of the library on a given language to be evaluated on a closed set of representative strings. The testsuite is constructed to test all grammatical constructions, as well as all ungrammatical constructions that can be generated by making one or two changes (e.g. in ordering, the presence or location of marking morphemes) to a grammatical example. The choices file represents the particular combination of features in the language being modeled and serves as input to the grammar-producing code. By attempting to parse the testsuite with the grammar generated by the customization system given the choices file, one can measure both the grammar’s coverage over grammatical sentences and its overgeneration. Sentences that are correctly parsed by the grammar must also be validated by hand to ensure that their parse trees are those expected and their corresponding semantic representations appear as intended.

## 6.1 *Pseudo-languages*

A pseudo-language is made for every possible combination of the following features, making 50 base pseudo-languages, all of which have possessor-first order:

Possessor status:

Modifier

Specifier

Possessor marking:

None

Affix (agreeing and non-agreeing)

Non-affix (agreeing and non-agreeing)

Possessum marking:

None

Affix (agreeing and non-agreeing)

Non-affix (agreeing and non-agreeing)

In addition, some of the above 50 pseudo-languages are altered to cover the following phenomena as well:

Order:

Possessor-first

Possessum-first

Pronouns:

Affix

Non-affix

Possessum marked (agreeing and non-agreeing)

Possessum unmarked

Constraints on the possessor:

Case

In total, 74 pseudo-language tests were constructed, which were incorporated into the Grammar Matrix repository as regression tests. Table 6.1 shows all of these pseudo-languages, the number of grammatical and ungrammatical examples in each, and the percentage of coverage and overgeneration for each.<sup>1</sup>

---

<sup>1</sup>Note that the name of each pseudo-language is composed of abbreviations that describe its features.

Table 6.1: Pseudo-language results

	# gram- matical	# ungram- matical	% coverage	% over- gen.
adnom-poss-mod-dep-aff-free-wo	2	6	100	0
adnom-poss-mod-dep-aff-head-aff-hf	1	6	100	0
adnom-poss-mod-dep-aff-head-aff-hf-agr-dep	1	3	100	0
adnom-poss-mod-dep-aff-head-aff-hf-agr-head	2	3	100	0
adnom-poss-mod-dep-aff-head-aff-hf-agr-mut	2	3	100	0
adnom-poss-mod-dep-aff-head-aff-hi	1	5	100	0
adnom-poss-mod-dep-aff-hf	1	5	100	0
adnom-poss-mod-dep-aff-hf-agr	1	1	100	0
adnom-poss-mod-dep-aff-hf-block-pron	2	3	100	0
adnom-poss-mod-dep-aff-hf-pron	1	5	100	0
adnom-poss-mod-dep-aff-hf-pron-aff-agr	1	1	100	0
adnom-poss-mod-dep-aff-hf-pron-aff-mod	1	2	100	0
adnom-poss-mod-dep-aff-hf-pron-agr	1	1	100	0
adnom-poss-mod-dep-aff-hi	1	8	100	0
adnom-poss-mod-dep-non-aff-free-wo	2	8	100	0
adnom-poss-mod-dep-non-aff-head-aff-free-wo	1	6	100	0
adnom-poss-mod-dep-non-aff-head-aff-hf	1	8	100	0
adnom-poss-mod-dep-non-aff-head-aff-hf-agr-dep	1	3	100	0
adnom-poss-mod-dep-non-aff-head-aff-hf-agr-head	1	1	100	0
adnom-poss-mod-dep-non-aff-head-aff-hf-agr-mut	1	7	100	0
adnom-poss-mod-dep-non-aff-head-aff-hf-sov	1	11	100	0
adnom-poss-mod-dep-non-aff-head- aff-hf-sov-marker-left	1	8	100	0
adnom-poss-mod-dep-non-aff-head-aff-hi-sov	1	12	100	0
adnom-poss-mod-dep-non-aff-head-non-aff-hf- agr-mut	1	3	100	0
adnom-poss-mod-dep-non-aff-head- non-aff-hf-sov-right-marker	1	4	100	0
adnom-poss-mod-dep-non-aff-hf	1	6	100	0
adnom-poss-mod-dep-non-aff-hf-agr	1	1	100	0
adnom-poss-mod-dep-non-aff-hf-block-head- pron-only	3	1	100	0
adnom-poss-mod-dep-non-aff-hf-block-pron	2	3	100	0

The most common abbreviations are as follows: ‘mod’ and ‘spec’ indicate modifier-like or specifier-like possessors; ‘hf’ and ‘hi’ indicate head-final or head-initial possessive constructions; and a string such as ‘dep-aff’ indicates that the dependent (possessor) is marked by an affix.

adnom-poss-mod-dep-non-aff-hi	1	10	100	0
adnom-poss-mod-head-aff-free-wo	2	8	100	0
adnom-poss-mod-head-aff-hf	1	6	100	0
adnom-poss-mod-head-aff-hf-agr	1	2	100	0
adnom-poss-mod-head-aff-hf-constraint	1	1	100	0
adnom-poss-mod-head-aff-hi	1	6	100	0
adnom-poss-mod-head-non-aff-hf	1	7	100	0
adnom-poss-mod-juxt	1	5	100	0
adnom-poss-mod-pron-head-aff-agr	2	4	100	0
adnom-poss-mod-pron-head-non-aff-agr	1	6	100	0
adnom-poss-spec-dep-aff-free-wo	2	6	100	0
adnom-poss-spec-dep-aff-head-aff-hf	1	6	100	0
adnom-poss-spec-dep-aff-head-aff-hf-agr-dep	1	3	100	0
adnom-poss-spec-dep-aff-head-aff-hf-agr-head	2	3	100	0
adnom-poss-spec-dep-aff-head-aff-hf-agr-mut	1	3	100	0
adnom-poss-spec-dep-aff-head-non-aff-hf	1	8	100	0
adnom-poss-spec-dep-aff-head-non-aff-hf-agr-dep	1	3	100	0
adnom-poss-spec-dep-aff-hf	1	4	100	0
adnom-poss-spec-dep-aff-hf-agr	1	2	100	0
adnom-poss-spec-dep-aff-hf-block-pron	2	2	100	0
adnom-poss-spec-dep-aff-hf-pron	1	5	100	0
adnom-poss-spec-dep-aff-hf-pron-aff	1	2	100	0
adnom-poss-spec-dep-aff-hf-pron-agr	1	1	100	0
adnom-poss-spec-dep-aff-hi	1	8	100	0
adnom-poss-spec-dep-non-aff-free-wo	2	6	100	0
adnom-poss-spec-dep-non-aff-head-aff-free-wo	2	8	100	0
adnom-poss-spec-dep-non-aff-head-aff-hf	1	6	100	0
adnom-poss-spec-dep-non-aff-head-aff-hf-agr-dep	1	3	100	0
adnom-poss-spec-dep-non-aff-head-aff-hf-agr-head	1	1	100	0
adnom-poss-spec-dep-non-aff-head-aff-hf-agr-mut	2	3	100	0
adnom-poss-spec-dep-non-aff-head-non-aff-free-wo	2	9	100	0
adnom-poss-spec-dep-non-aff-head-non-aff-hf	1	6	100	0
adnom-poss-spec-dep-non-aff-head-non-aff-hf-agr-dep	1	2	100	0
adnom-poss-spec-dep-non-aff-hf	1	5	100	0
adnom-poss-spec-dep-non-aff-hf-agr	1	1	100	0
adnom-poss-spec-dep-non-aff-hf-block-head-pron-only	3	1	100	0
adnom-poss-spec-dep-non-aff-hf-block-pron	2	2	100	0
adnom-poss-spec-dep-non-aff-hf-constraint	1	1	100	0

adnom-poss-spec-head-aff-free-wo	2	6	100	0
adnom-poss-spec-head-aff-hf	1	4	100	0
adnom-poss-spec-head-aff-hf-agr	1	2	100	0
adnom-poss-spec-head-aff-hf-block-head-pron-only	3	1	100	0
adnom-poss-spec-head-non-aff-hf	1	5	100	0
adnom-poss-spec-head-non-aff-hf-block-head-pron-only	3	1	100	0
adnom-poss-spec-head-non-aff-hf-constraint	1	1	100	0
adnom-poss-spec-juxt	1	6	100	0
adnom-poss-spec-juxt-block-head-pron-only	2	1	100	0
adnom-poss-spec-juxt-block-pron	1	2	100	0
adnom-poss-spec-juxt-constraint	1	1	100	0
adnom-poss-spec-pron-head-aff-agr	3	2	100	0
adnom-poss-spec-dep-aff-two-strat-order-var	2	2	100	0
adnom-poss-dep-aff-two-strat-mod-spec-var	2	2	100	0

## 6.2 Illustrative languages

The languages selected for illustrative testing are Finnish (fin, Uralic), Yoruba (yor, Niger-Congo), Ancient Greek (grc, Indo-European), Fijian (fij, Austronesian), and Japanese (jpn, Japonic). These languages were chosen as a fair representation of the breadth of phenomena covered by this library, as well as a reasonably diverse sampling from the world’s languages, with no two languages representing the same language family. In this section, I give a brief description of relevant facts for each language, how those facts were modeled, and the coverage and overgeneration when a grammar is produced and used to parse the testsuite. In all cases, where I report that a grammatical string was parsed, its parse tree and corresponding semantic representation were hand-checked to confirm they were properly formed. Both testsuites and grammars for each of these languages can be found in the Grammar Matrix repository, though the examples shown in a given testsuite may have simplified orthography, and are generally author-generated, rather than being attested forms.

### 6.2.1 Finnish

Finnish is a Uralic language with flexible word order, with SVO ordering being the least marked. Finnish has a robust nominative-accusative case system and no marking of gender. No Finnish nouns require determiners, though they may optionally appear. The following possessive strategies are found in Finnish:

1. **Possessor is marked with genitive case; possessum is unmarked:**

Hanna-n auto  
 Hanna-GEN car  
 ‘Hanna’s car’ [fin] (Karlsson, 2015, 117, gloss mine)

2. **Possessor is a pronoun; possessum is marked with an agreeing affix:**

minun velje-ni  
 my brother-POSS.1SG  
 ‘my brother’ [fin] (Karlsson, 2015, 125, gloss mine)

3. **Possessor is a pronominal affix on the possessum:**

velje-ni  
 brother-POSS.1SG  
 ‘my brother’ [fin] (Karlsson, 2015, 125, gloss mine)

The three strategies outlined above are analyzed as follows. Strategy (1) is analyzed as a specifier-like unmarked construction in which the possessor is constrained to have the feature [ CASE *gen* ]. Strategy (2) is analyzed as a class of specifier-like possessor pronouns which trigger person and number agreement in the possessum. Strategy (3) is analyzed as a class of specifier-like affixal possessor pronouns. The above analyses achieve 100 percent coverage and zero percent overgeneration

Strategies (2) and (3) could perhaps be better analyzed as two manifestations of the same strategy, with the overt possessor dropped in (3). Indeed, there is no particular evidence that strategies like (3) differ from strategies like (2) in any way other than the presence of the overt possessor; they are largely parallel to other sorts of pro-drop constructions. However, the analyses of (2) and (3) in the library as presently configured would differ too greatly to relate them in this way, even if the questionnaire were adapted to allow the user to enter the strategy in this fashion. In (2), the needed possessive semantics are carried by the pronoun

*minun*, while in (3), they’re carried by the affix *-ni*. This makes meshing these constructions into one difficult. Instead, they are analyzed as fully distinct strategies, and the suffix *-ni* is treated as two homophonous suffixes.

In all cases where a grammatical instance is described as ‘covered’ (here and elsewhere), I have verified that it maps to the correct semantic representation. An example of a correct semantic representation of the sentence in example (80) is shown in figure 6.1, as produced by the grammar.

- (80) *tytö-n koira tuli*  
 girl-GEN dog.NOM come.3SG.PST  
 ‘the girl’s dog came’ [fin] (Author-generated example, based on Karlsson (2015))

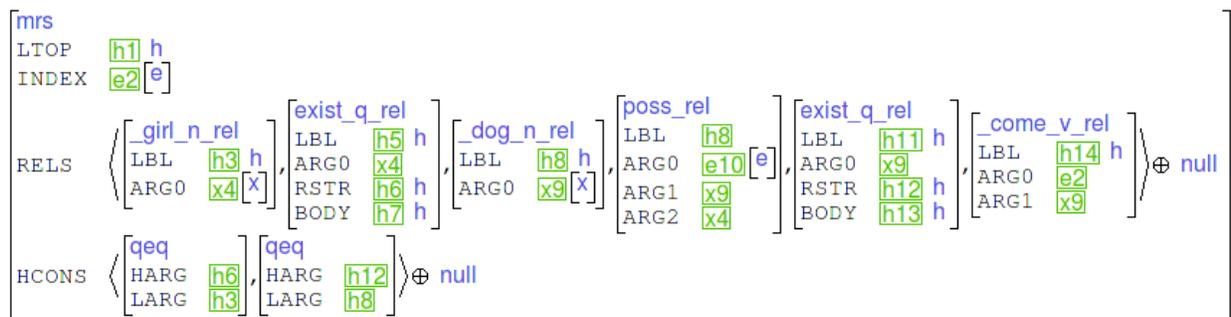


Figure 6.1: MRS representation of the Finnish sentence *tytön koira tuli* ‘the girl’s dog came.’

### 6.2.2 Yoruba (*yor*)

Yoruba is a Niger-Congo language spoken in Benin and Nigeria. It has strict subject-verb-object word order, and does not have grammatical gender or case. There are no determiners in Yoruba (Bámgbósé, 2000). There are two distinct possessive strategies in Yoruba: the first is an unmarked possessum-first construction (see (81)); the second is a possessum-first construction wherein the possessor is marked with a clitic (see (82)). In both cases, the possessor is analyzed as specifier-like, though the lack of determiners makes it difficult to

determine if this is appropriate, since the clearest test for a possessor’s specifier-hood is whether it can co-occur with a determiner. In both cases, the possessor can be a pronoun or a common noun; no specialized classes of possessor pronouns exist.

(81) aṣọ ẹbí  
 dress family  
 ‘the family’s dress’ [yor] (Bámgbósé, 2000, 110, gloss mine)

(82) aṣọ t=ẹbí  
 dress POSS=family  
 ‘the family’s dress’ [yor] (Bámgbósé, 2000, 110, gloss mine)

In order to model these facts, I implement two distinct strategies in the questionnaire, one unmarked, and one marked. A testsuite is constructed to exercise both these possibilities, with both common nouns and pronouns in the possessor slots. Corresponding ungrammatical examples are generated in the testsuite by varying the order of constituents, as well as by dropping or doubling possessive markers.

No bugs were discovered in the construction and running of this test; the testsuite showed 100 percent coverage with zero percent overgeneration.

### 6.2.3 Ancient Greek (grc)

Ancient Greek is an extinct Indo-European language and the ancestor to Modern Greek. It has free major constituent order, and has case, number, person, and gender. Determiners exist and are optional in some contexts (Mastronarde, 2013). There are two major possessive patterns to be modeled. First, nouns marked with the genitive case can act as modifier-like possessors:

(83) he:                    to                    patròs                    oikía  
 the.F.SG.NOM the.M.SG.GEN father(M).SG.GEN house(F)SG.NOM  
 ‘the father’s house’ [grc] (Goodwin, 1894)

Second, there is a series of possessor pronouns that agree with the possessum in number and case:

- (84) ho                    emó-s    filo-s  
 the.M.SG.NOM my-NOM friend-NOM  
 ‘my friend’ [grc] (Smyth, 1920, gloss mine)
- (85) ton                    emó-n    filo-n  
 the.M.SG.ACC my-ACC friend-ACC  
 ‘my friend’ [grc] (Smyth, 1920, gloss mine)

The first strategy, where a common noun that is marked with genitive case acts as a possessor, is analyzed as an unmarked possessive construction where the possessor is constrained to be [ CASE *gen* ] (as outlined in section 4.2.1 above). The second case is implemented as a series of possessor pronouns which agree with the possessum in number and case.

The test suite modeled both these strategies, including ungrammatical examples, with incorrect ordering and markers doubled or missing. The grammar produced by the library achieved 100 percent coverage with zero percent overgeneration.

Several extensions and fixes were made to the system in the course of modeling Ancient Greek. Most significantly, case agreement between possessor pronouns and the possessum had to be implemented. This is a sufficiently common pattern that it merited implementation, although, as noted in section 4.5.2 above, it was implemented only for modifier-like possessor pronouns, not specifier-like pronouns, and was not implemented at all for common noun possessive strategies.

#### 6.2.4 *Fijian (fij)*

Fijian is an Austronesian language with VOS word order. The variety under discussion here is Boumaa Fijian, as described by Dixon (1988). Case and gender are not marked in Fijian, though three numbers (singular, dual, and plural) exist. Determiners can optionally accompany any noun. Fijian has many possessive strategies, which are used for different classes of nouns. The correctness of a given possessive strategy depends on whether the possessor is a proper noun, a common noun, or a pronoun; and whether the possessum is ‘bound’ or ‘free’ (meaning, roughly, inalienably possessed or alienably possessed) (Dixon,

1988). These strategies are given by Dixon as follows:

1. **Proper possessor, bound possessum:** clitic marking on possessum:

a liga=i        Jone  
the hand=POSS John

‘John’s hand’ [fij] (adapted from Dixon, 1988, 120, gloss mine)

2. **Proper possessor, free possessum:** clitic marking on classifier of possessum:

a wagona me=i        Jone  
the kava    CLSFR=POSS John

‘John’s kava’ [fij] (adapted from Dixon, 1988, 120, gloss mine)

3. **Pronoun possessor, bound possessum:** possessor pronoun appears as clitic on possessum:

a liga=na  
the hand-3SG.POSS

‘his/her hand’ [fij] (Dixon, 1988, 121, gloss mine)

4. **Pronoun possessor, free possessum:** possessor pronoun appears as clitic on classifier of possessum:

a me=na                wagona  
the CLSFR=3SG.POSS kava

‘his/her kava’ [fij] (Dixon, 1988, 121, gloss mine)

5. **Common possessor, bound or free possessum:** possessor marked with adposition:

tuuraga ni    vanua  
chief    POSS place

‘the chief of the place’ [fij] (Dixon, 1988, 121, gloss mine)

Note that while Dixon calls the marker in strategies (1) and (2) an affix, I have reanalyzed it as a clitic. The fact that it can attach either to the possessum or to the classifier that marks the possessum indicates that it is attaching to the phrase, not the word.

Of these constructions, (2) and (4) cannot be modeled in the Grammar Matrix customization system directly, since no library to model classifiers in any language has been built. If

such a library is built in the future, these constructions could in theory be handled at that point. For the present, the choices file and testsuite cover only strategies (1), (3), and (5).

In the case of strategy (3), a class of left-attaching possessor pronouns with modifier-like attachment is created. The distinction between a bound and free possessum cannot be fully modeled, because the library currently only allows the user to specify feature constraints on the possessor, not the possessum. This limitation was imposed because, in many languages, constraints on the possessum (such as case) are determined by the role the possessive phrase plays in the whole sentence, rather than by the type of possessor. Additionally, modeling this type of inalienable/alienable distinction was not within the primary scope of this thesis, as noted in section 3.3.3.2. In some cases, existing architecture from other libraries allows the user to model this distinction directly; in Fijian, it falls short. As a result, the library achieves 100 percent coverage over two grammatical instances of this phenomenon, but also overgenerates by parsing one of three ungrammatical instances. However, because inalienable possession is out of scope, I do not attempt to rectify this overgeneration at the present time.

Strategy (5) can be modeled by implementing a possessive strategy wherein the possessor is marked by a left-attaching adposition. In this case, the inalienable/alienable distinction can be fully modeled. The possessor can be constrained in the questionnaire to be a common noun, and no constraints appear on the possessum. The library achieves 100 percent coverage over this phenomenon with no overgeneration

Finally, strategy (1) poses a problem. In the desired analysis, the possessum-marking clitic *i* takes the possessum as a complement, attaching via a head-final *comp-head* phrase. This complex of possessum and clitic then takes the possessor as its complement in a head-initial *head-comp* phrase. An issue arises because the *comp-head* phrase has the constraint [ INIT – ] on its head daughter, indicating that it is a head-final rule. Since INIT is a head feature, it propagates up to the parent of *comp-head*. This means that the complex of possessum and clitic has the value [ INIT – ], and therefore is not allowed in the head daughter role of the head-initial *head-comp* rule. This means that, even if the clitic *i* does not have a constrained value for INIT, it still cannot participate in both the *head-comp* and

the *comp-head* rules. In general, it is not possible for the order of possessum and marker to differ from the order of possessum and possessor in cases where the possessive phrase is a head-complement phrase. As a result, the grammar achieves zero percent coverage of this phenomenon if I implement this analysis.

However, another analysis of strategy (1) makes it possible to model the facts in Fijian. The evidence Dixon (1988) provides for the clitic (or in his terms, affix)<sup>2</sup> *i* being attached to the possessum is phonological: in Fijian, the diphthongization of two adjacent vowels which is observed *liga=i* in strategy (1) occurs only when those vowels occur in a single phonological word (Dixon, 1988, 15). However, a clitic's phonological host need not be the same as its syntactic host (Klavans, 1985). Absent non-phonological evidence to the contrary, it is possible to analyze the clitic *i* as attaching syntactically to the possessor, rather than to the possessum. Note that this analysis is not strongly supported by the data; rather, it is simply not contradicted by it. With this analysis in place, it is possible to analyze strategy (1) in largely the same terms as (5): a left-attaching clitic marks the possessor, with the possessum remaining unmarked. This analysis achieves 100 percent coverage, with, again, overgeneration in the form of two ungrammatical sentences that parse because of deficiencies in modeling inalienable possession.

Over all strategies, this analysis achieves 100 percent coverage. Furthermore, because all the cases of overgeneration in Fijian arose from the inability of the library to model inalienable possession fully, which was not within the primary scope of the library, for present purposes, the library is considered to have achieved zero percent overgeneration on Fijian.

### 6.2.5 Japanese (*jpn*)

Japanese is an SOV language with a nominative-accusative case system and no gender marking. Japanese has only one possessive strategy, wherein the possessor is marked by the postposition *no* (McGloin, 2014, 28). Common nouns and pronouns can both act as possessors:

---

<sup>2</sup>Note that if I adopt Dixon (1988)'s analysis and call this marker an affix, the analysis becomes straightforward, since the head-complement rule would not be involved in attaching the marker to the possessum.

- (86) sensee no hon  
 teacher POSS book  
 ‘teacher’s book’ [jpn] (McGloin, 2014, 28, gloss mine)

This construction is analyzed as a specifier-like possessor with non-affixal marking. The grammar generated achieves 100 percent coverage and zero percent overgeneration.

### 6.2.6 Summary

The library in its current state is able to model all the in-scope the phenomena in Finnish, Yoruba, Ancient Greek, Fijian, and Japanese. The number of grammatical and ungrammatical examples in each language’s testsuite is given, as well as the percentage of coverage and overgeneration. Note that the overgeneration in the case of Fijian is due to out-of-scope phenomena, and so is not counted as genuine overgeneration.

Table 6.2: Illustrative language results

	# gram- matical	# ungram- matical	% coverage	% over- generation
Finnish [fin]	8	13	100	0
Yoruba [yor]	4	17	100	0
Ancient Greek [grc]	11	6	100	0
Fijian [fij]	7	7	100	28.6
Japanese [jpn]	2	13	100	0

### 6.3 Held-out languages

In order to evaluate the coverage of the library, I conducted tests on a set of five randomly selected languages, with the implementation frozen at its current stage of development. All errors in the adnominal possession library discovered at this point are reported below. These languages were selected from a list of 2240 assorted language resources by using a random-number generating script. Languages that were from the same family as a previously selected language or which lacked sufficient resources were discarded. The selected languages were

Sölring North Frisian (frr, Indo-European), Coatlan-Loxicha Zapotec (ztp, Otomanguean), Plang (blr, Austro-Asiatic), Washo (was, isolate), and Georgian (kat, Kartvelian).

### 6.3.1 *Sölring North Frisian*

Sölring North Frisian is a variety of Frisian, a constellation of Germanic languages spoken in the Netherlands and Germany. The Frisian languages are very close relatives of English and as such, Sölring North Frisian shares many features with English: it has SVO basic word order, only the vestiges of a case system, and distinguishes two genders and two numbers (Lasswell, 1998).

In Sölring North Frisian, there is a class of possessor pronouns, as well as two strategies for encoding possessive phrases with a common noun possessor. The possessive pronoun is explicitly described as a determiner by Lasswell (1998), suggesting that the specifier-like analysis is most appropriate. Its use is shown in (87) below:

- (87) sin                    hingst  
       3SG.MASC.POSS horse  
       ‘his horse’ [frr] (Lasswell, 1998, 257, gloss adapted)

These possessor pronouns are entered into the customization system as a set of specifier-like possessor pronouns. They are constrained to be left-attaching, with no agreement features present.

Of the two possessive strategies, the first is not used outside of the written language or set expressions today. It consists of an affix *-s* that appears on the possessor, a pattern with obvious correlates in other Germanic languages. An example of this pattern appearing in a set expression is shown in (88) below.

- (88) ön Got-s        noom  
       in God-POSS name  
       ‘in God’s name’ [frr] (Lasswell, 1998, 257, gloss adapted)

This strategy is entered into the customization system as a specifier-like strategy where the

possessor is marked with an affix. The possessor is constrained to be left-attaching and no other feature constraints are added.

The second possessive strategy is more commonly used in the spoken language. It consists of a possessum, followed by an agreeing possessor pronoun and the possessor. An example of this pattern is shown in (89) below:

- (89) di man sin                                    hingst  
       the man 3SG.MASC.POSS horse  
       ‘the man’s horse’ [frr] (Lasswell, 1998, 257, gloss adapted)

The analysis of the word *sin* as a possessive pronoun in this context is not readily producible by the library. Instead, rather than analyzing this *sin* as identical to the possessor pronoun *sin*, I analyze this word as a non-affixal marker on the possessum, which happens to agree in person and number with the possessor. The marker is constrained to attach to the possessum from the left, and the possessor is constrained to attach to the possessum from the left. The fact that the possessum *hingst* does not take an article in (89) means that this construction should be analyzed as specifier-like.

These analyses achieved 100 percent coverage. However, two bugs in the library led to 33.3 percent overgeneration. The bugs were as follows: as noted in chapter 4, the head-complement rule must explicitly copy up the head-daughter’s POSSESSUM value to the parent. The code that added this was incorrectly specific to only modifier-like constructions, and so this constraint was not added in this case. Without this constraint, the possessor could not access the POSS-AGR features of the possessum and so sentences with incorrect agreement patterns were parsed when they should have been rejected. Secondly, the possessum-marking word ought to constrain the possessum to be both [ POSSESSOR *nonpossessive* ] and [ POSSESSUM *nonpossessive* ]. Again, the code that added this constraint was too specific and only added this constraint to a subset of possessum markers. Both of these bugs were deficiencies in the implementation, not failures of the analysis, and were addressed by post-evaluation changes (see section 6.4) by generalizing the relevant sections of code.

### 6.3.2 Coatlan-Loxicha Zapotec

Coatlan-Loxicha Zapotec is an Otomanguean language spoken in the state of Oaxaca in southern Mexico. It has VSO basic word order and a nominative-accusative case system (Beam de Azcona, 2004, 15). There are two primary possessive strategies, one that is used when the possessum is an inalienably possessed noun (such as a noun referring to a family member or body part), and another when the possessum is an alienably possessed noun. Both these constructions allow nominative pronouns to function as the possessor and no separate possessor pronouns exist.

Inalienably possessed nouns are marked by the possessive prefix *x-* and the possessor attaches to their right. The possessor must be in the nominative case, and may be a pronoun (which takes the form of a clitic):<sup>3</sup>

- (90) a. *x-áb ya*  
           POSS-bark tree  
           ‘the tree’s bark’ [ztp] (Beam de Azcona, 2004, 302)
- b. *x-na<sup>7</sup>=s*  
           POSS-mother=1EXCL  
           ‘our mother’ [ztp] (Beam de Azcona, 2004, 302)

In the case where the possessor is the first-person singular pronoun, the possessum acquires a final high tone, which changes its overall tone in accordance with the phonological rules of Coatlan-Loxicha Zapotec. This can be seen in the following examples, where the word *father* appears with a first-person singular and a second-person singular pronoun:

- (91) a. *x-ũz nâ*  
           POSS-father 1SG  
           ‘my father’ [ztp] (Beam de Azcona, 2004, 303)
- b. *x-ùz lô*  
           POSS-father 2SG  
           ‘your father’ [ztp] (Beam de Azcona, 2004, 303)

---

<sup>3</sup>All diacritics in (90) represent tone and register values. The superscript <sup>7</sup> represents glottalization, which is contrastive in Coatlan-Loxicha Zapotec.

In general, the Grammar Matrix does not model non-concatenative morphology directly, so modeling this as a mutation of the noun root *ũz* ‘father’ is not possible. However, if the tone variation is analyzed as a property of the possessive affix, rather than the root, this can be modeled as a very limited case of agreement, where the possessum agrees with a first-person singular possessor. This strategy is entered into the customization system as a specifier-like strategy with possessum marking (with agreement) and no marking on the possessor.

The second strategy in Coatlan-Loxicha Zapotec is used for alienably possessed nouns. In this case, the possessor still attaches to the right of the possessum, but it is the possessor which is marked and not the possessum. The possessor marker takes the form of an adposition *tě*, and again, the possessor may be a common noun or a pronoun:

- (92) *báy*      *tě=m*  
kerchief POSS=2SG  
‘his kerchief’ [ztp] (Beam de Azcona, 2004, 306)

The possessor marker undergoes a tonal mutation when followed by the first person singular pronoun, similar to the tonal mutation seen in the inalienable possessive construction. For this reason, the preposition *tě* that Beam de Azcona (2004) analyzes as marking the possessor must be analyzed as a postposition marking the possessum and agreeing with the possessor.<sup>4</sup> The analysis of *tě* as a preposition is not necessarily a widely accepted analysis among linguists studying Zapotecan languages, since the existence of the category of prepositions is itself disputed (Beam de Azcona, 2004, 2).

With the analysis implemented as described here, the grammar was able to achieve 100 percent coverage over five grammatical examples. However, the bugs noted in the case of Sölring North Frisian caused issues here again, leading to 5.6 percent overgeneration. Because the complement of the possessum-marking word was not constrained to be [ POSSESSUM *nonpossessive* ], cases where the possessum-marking word was doubled were

---

<sup>4</sup>Positing a postposition in a VSO language is typologically marked, making this analysis somewhat less supported.

incorrectly parsed.

Though it is not strictly in the scope of this thesis, it should be noted that the inalienable/alienable distinction in Coatlán-Loxicha Zapotec can also be successfully modeled. The inalienable possessive prefixes can be constrained to only appear on inalienably possessed nouns—taking advantage of the architecture of the morphotactics library, as outlined in chapter 2. So long as this prefix is marked as obligatory, inalienably possessed nouns will not be able to appear prefixless in alienable constructions.

### 6.3.3 *Plang*

Plang is an Austroasiatic language spoken in the Yunnan Province of China, and, since the 1970s, in parts of Myanmar and Thailand (Block, 1994, 1). Its morphology is highly isolating, as is common to languages spoken in Southeast Asia. It has SVO word order and no case or gender system. It does distinguish between singular, dual, and plural number (Block, 1994).

There are two possessive strategies in Plang, one that only admits unmarked pronouns as possessors and one that only admits common nouns as possessors. The first of these strategies consists is an unmarked possessive construction where the possessor—a pronoun which is not distinguished from the general personal pronouns in any way—attaches from the right. This construction is shown in (93) below:

- (93) yung iq  
       village 1PL  
       ‘our village’ [blr] (Block, 1994, 16)

This could be modeled in two different ways in the customization system. First, these pronouns could be entered as a series of possessor pronouns that just happen to coincide completely with personal pronouns. Second, this could be entered as a possessive strategy, with a user-defined feature corresponding to pronoun-hood required to be present on the possessor. Note that, although the feature PRON already exists, it is not made available to the user to specify directly as a required feature in the pull-down menus of the questionnaire.

This is an oversight that should be corrected. However, even though it is somewhat ungainly to add a second pronoun feature, it is certainly possible to do so and so to model the facts of Plang. For the purposes of this test, this second option—adding a new PRONOUN feature that is required to have a positive value on the possessor—was taken, though another analysis was possible.

The second possessive strategy in Plang consists of an unmarked possessum followed by a possessor which is marked by a pronoun. Though Block (1994) gives limited data (there is only one example of this sort of construction), it appears that this pronoun agrees in person and number with the possessum. This agreement is shown in (94), where the possessum is understood as plural (even though there is no plural inflection) because of the plural number of the pronoun *kiq*:

- (94) hmal            taqhuq            kiq  
 dry.rice.field grandfather.Chinese 3PL  
 ‘the fields of an old Chinese man’ (Block, 1994, 16)

This strategy is entered into the customization system as a right-attaching possessor which is marked by a right-attaching possessive marker which agrees with the possessum in person and number. The coincidence of the possessive marker with the possessive pronouns is not captured by this analysis, but the phenomena at hand are covered nonetheless.

This analysis achieves 100 percent coverage over 2 positive examples, with 12.5 percent overgeneration over 8 ungrammatical examples. This comes about because the pronouns which are analyzed as possessor-marking adpositions (such as *kiq* in (94) above) are constrained so as to take as complement any full NP which has the features [ POSSESSOR *non-possessive*] and [ POSSESSUM *nonpossessive*]. In most contexts, a possessive noun phrase has identical distribution to other noun phrases, and so the library has been consciously designed so as to make possessive NPs featurally identical to nonpossessive NPs from the point of view of other elements in the sentence. However, this makes it impossible to rule out constructions where the possessor-marking adposition *kiq* takes an entire possessive NP as its complement, such as (95):

- (95) \*[hmal            taqhuq                            kiq]<sub>np</sub> kiq  
       dry.rice.field grandfather.Chinese 3PL    3PL

This is a deficiency of the analysis which, unlike the bugs previously mentioned, has no obvious immediate solution to be implemented. Its solution is left as future work.

#### 6.3.4 *Washo*

Washo is an isolate language spoken by “several dozen” people in the Lake Tahoe region of California (Simons and Fennig, 2018). It has SOV basic word order and a case system that is confined to a few oblique cases. Though there is a distinction between singular and plural number in Washo, no number distinctions are made in the personal pronoun paradigm (Jacobsen, 1964).

Washo has a class of nouns that must appear with a possessor (expressed as a pronominal affix). Jacobsen (1964) refers to these as ‘possessed’ nouns, in contrast to two other classes: ‘restricted’ nouns, which must appear with either a possessor or an absolutive affix, and ‘unrestricted’ nouns, which may appear either with or without a possessor. An example of a restricted noun appearing with a possessor prefix<sup>5</sup> is shown below:

- (96) l<sup>e</sup>-iheb  
       1.POSS-head  
       ‘my head’ [was] (Jacobsen, 1964, 411, gloss mine)

When any noun appears with a possessor prefix, it can also be accompanied by an overt possessor in the form of a pronoun or a common noun:

- (97) l<sup>e</sup>i l<sup>e</sup>-iheb  
       1 1.POSS-head  
       ‘my head’ [was] (Jacobsen, 1964, 412, gloss mine)

These two behaviors — possessive pronoun affixes appearing with and without overt possessors — are added to the customization system as distinct entities. When the possessive

---

<sup>5</sup>Note that the possessor prefixes may undergo drastic phonologically-conditioned allomorphy, which is not demonstrated by the examples given here. These examples were selected to suppress this for the purposes of this discussion.

pronoun affix appears alone, it is part of a class of possessive pronouns; when it appears with an overt possessor, it is a possessive strategy wherein the possessum is marked and shows person agreement with the possessor. In both cases, the possessive affixes are added to different position classes that apply variously to possessed, restricted, and unrestricted nouns. Each of these position classes has the correct settings for obligatoriness to ensure the desired behavior: the prefix is obligatory for possessed nouns, obligatory but possible to supplant with an absolutive prefix for restricted nouns, and optional for unrestricted nouns.

In addition to these strategies, possession can be expressed by means of an unmarked construction. Because only unrestricted nouns can be unmarked, the possessum must be an unrestricted noun. Presumably a properly marked possessed or restricted noun could act as possessor, but Jacobsen (1964) does not offer any clarification on this point.

- (98) géwe máyap  
 coyote paw  
 ‘coyote’s paw’ [was] (Jacobsen, 1964, 426)

This last construction is implemented as a possessor-initial unmarked construction. Because it is not currently possible to add feature constraints to the possessum, some nouns that are not unrestricted will appear in the possessum role of these constructions inappropriately, leading to 9.1 percent overgeneration. Since modeling inalienable possession is not in the primary scope of this thesis, this deficiency is not considered genuine overgeneration.

This analysis achieves 100 percent coverage and zero percent overgeneration over 18 total instances, including grammatical and ungrammatical examples.

### 6.3.5 *Georgian*

Georgian is a Kartvelian language spoken primarily in the Republic of Georgia. Georgian has a split-ergative case system, where most sentences have nominative-accusative alignment, while those with main verbs in the aorist or optative tenses have ergative-absolutive alignment (Nash, 2017, 4). Since ergativity is so restricted in Georgian, for the purposes of this test, Georgian is treated as a nominative-accusative language. Georgian makes no

distinctions in gender and has both a singular and a plural. Word order in Georgian is free, with the SOV and SVO orders being least pragmatically marked (Hewitt, 1995).

Georgian possesses a single possessive strategy for common noun possessors and a single set of possessor pronouns. The possessive strategy consists of a left-attaching possessor marked by the genitive case, as shown in (99) below.

- (99) sakhl-is    gasaghebi  
       house-GEN key  
       ‘the house’s key’ [kat] (Kiziria, 2008, 115, gloss mine)

This is entered into the customization system as an unmarked possessive construction, where the possessor is constrained to have genitive case. There are few strong arguments in favor of either a specifier-like or a modifier-like analysis, but a specifier-like analysis was chosen because the possessor attaches from the left side, like optional determiners do.

The possessor pronouns in Georgian attach from the left side and agree with the possessum in case. Their behavior can be seen in the examples shown in (100) below:

- (100) shen-i            shvil-i  
       2SG.POSS-NOM child-NOM  
       ‘your child’ [kat] (Hewitt, 1995, 86)
- chem-s            tav-s    shevek’itkhe  
       1SG.POSS-DAT self-DAT asked  
       ‘I asked myself’ [kat] (Hewitt, 1995, 86)

In order to model the case agreement between possessor pronoun and possessum, these must be analyzed as modifier-like possessor pronouns. They are entered into the customization system as a series of modifier-like, left-attaching possessor pronouns that show case agreement.

This analysis achieved 100 percent coverage and zero percent overgeneration over 12 total instances, 8 grammatical and 4 ungrammatical.

### 6.3.6 Summary

All tests conducted on held-out languages achieved 100 percent coverage. The tests conducted on Georgian (kat) and Washo (was) also achieved zero percent overgeneration. However, due to several bugs in the implementation of the analysis given in chapter 4, the grammars produced by the library for Sölring North Frisian (frr), Coatlan-Loxicha Zapotec (ztp), and Plang (blr) all produced overgeneration. While there are obvious, readily implementable solutions for the overgeneration observed in Sölring North Frisian and Coatlan-Loxicha Zapotec, the overgeneration in Plang (blr) does not have any such obvious solution given the current analysis. The results of these tests are shown in table 6.3 below:

Table 6.3: Held-out language results

	# gram- matical	# ungram- matical	% coverage	% over- generation
Sölring North Frisian [frr]	4	15	100	33.3
Coatlan-Loxicha Zapotec [ztp]	5	19	100	5.6
Plang [blr]	2	8	100	12.5
Washo [was]	6	12	100	9.1 <sup>6</sup>
Georgian [kat]	8	4	100	0

## 6.4 Post-evaluation changes

After running all held-out tests, the bugs noted above were corrected. First, the head-complement rule was made to copy up the value of the POSSESSUM feature in all cases, instead of just a limited subset. Second, all possessum-marking words were made to constrain the possessum to be both [ POSSESSOR *nonpossessive* ] and [ POSSESSUM *nonpossessive* ]. Generalizing both of these constraints eliminated overgeneration in the Sölring North Frisian and Coatlan-Loxicha Zapotec test suites.

In addition to the changes made to address bugs found in testing, some other fixes were

---

<sup>6</sup>This is overgeneration solely over out-of-scope phenomena, and so is not counted as genuine overgeneration.

added for bugs were identified by readers. First, as noted in section 4.2.2, the head-modifier rule also must be augmented to pass up the POSSESSUM value of the head daughter in languages with specifier-like possessive constructions. This allows a possessum to still be marked as a possessum even when it is modified by an adjective. Second, the specifier-like possessor pronoun affix was altered so that its parent did not have the feature [ SPR ⟨ ⟩ ], but rather had the feature [ SPR ⟨ [ OPT + ] ⟩ ], as discussed in section 4.4.2.

### **6.5 Summary**

In this description, I have described the evaluation of this library over a set of pseudo-languages, as well as a set of five illustrative languages and five held-out languages. In all cases, the library was able to achieve 100 percent coverage of the phenomena at hand. There was no overgeneration in the tests of pseudo-languages and illustrative languages, but a few bugs led to some overgeneration in the case of tests on held-out languages.

## Chapter 7

### CONCLUSION

This thesis describes the development of an adnominal possession library for the Grammar Matrix customization system. The successful implementation of this library adds to the usefulness of the Grammar Matrix as a tool for linguists and others interested in rapidly developing precision HPSG grammars. Being able to readily model a typologically widespread phenomenon such as adnominal possession expands the utility of the Grammar Matrix customization system in modeling almost any language.

In this thesis, I have described the process of development, beginning in chapter 2 with a discussion of the architecture of the Grammar Matrix and its theoretical underpinnings (including Head-driven Phrase Structure Grammar and Minimal Recursion Semantics). In chapter 3, I give an outline of the typological variation seen in adnominal possessive phrases in the world's languages. I then give a description of the analysis I posit for this phenomenon in chapter 4. Finally, I describe the implementation of this analysis in a Grammar Matrix customization system library (chapter 5) and the evaluation (chapter 6) of the completed library.

In addition to producing a working Grammar Matrix library, in the process of completing this thesis I have gleaned some information about related topics as well, including the typology of possession and the importance of types in grammar engineering. First, I have demonstrated that all marked possessive constructions can be modeled using preexisting phrase structure rules. It is common in the typological literature to see possessors described as simply a subtype of specifiers or modifiers (cf. Lasswell 1998; Plank 1992), but exactly how comparable these categories were was not clear. By implementing an analysis for both these types of phrases using only phrase structure rules that already existed in the Gram-

mar Matrix, I have provided evidence in support of the idea that possessors are genuinely subtypes of specifiers and modifiers, and not a distinct category unto themselves. Second, I was able use types defined by previous developers of the Grammar Matrix, which turned out to provide significant advantages in the multi-lingual grammar engineering context. For example, the type PNG could be used as the value of my newly defined POSS-AGR feature, which allowed me to abstract away from the details of which person, number, and gender categories existed in a given language. This shows that types not only provide efficiency as a monolingual level (Flickinger, 2000), they also add efficiency to cross-linguistic grammar engineering. These conclusions were not the primary object of producing a Grammar Matrix library, but they do demonstrate that the benefits of such work extend beyond the production of the library itself.

While the library presently covers the phenomenon of adnominal possession as defined here, there are several possible directions for further development. First, inalienable possession, which is well represented in the illustrative and held-out language sets described in chapter 6, is only partially covered by the library. Relatively minor additions to the adnominal possession questionnaire would allow for the user to model these constructions more completely. Second, the phenomenon of case agreement between possessor pronouns and their heads is handled only in the limited case of modifier-like possessor pronouns. A useful addition to this library would be an expanded analysis of case agreement that covered specifier-like possessor pronouns. While there are still directions for future work, this thesis provides a foundation on which to base such work.

## BIBLIOGRAPHY

- Edith Aldridge. *Ergativity and word order in Austronesian languages*. PhD thesis, Cornell University, 2004.
- Ayò Bámbósé. *A Grammar of Yoruba*. West African language monograph series. Cambridge University Press, 2000.
- Rosemary Grace Beam de Azcona. *A Coatlan-Loxicha Zapotec grammar (Mexico)*. PhD thesis, University of California, Berkeley, 2004.
- Emily M. Bender and Dan Flickinger. Rapid prototyping of scalable grammars: Towards modularity in extensions to a language-independent core. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing IJCNLP-05 (Posters/Demos)*, Jeju Island, Korea, 2005.
- Emily M. Bender, Dan Flickinger, and Stephan Oepen. The Grammar Matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In J. Carroll, N. Oostdijk, and R. Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei, Taiwan, 2002.
- Emily M. Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. Grammar customization. *Research on Language & Computation*, 8(1):23–72, 2010. ISSN 1570-7075. URL <http://dx.doi.org/10.1007/s11168-010-9070-1>. 10.1007/s11168-010-9070-1.
- Karen Louise Block. Discourse grammar of first person narrative in Plang. Master’s thesis, University of Texas at Arlington, 1994.

- Kaja Borthen and Petter Haugereid. Representing referential properties of nominals. *Research on Language and Computation*, 3(2):221–246, Jul 2005. URL <https://doi.org/10.1007/s11168-005-1297-x>.
- Christian Brothers. *Irish grammar*. M.H. Gill, 1910.
- A. Copestake, D. Flickinger, C. Pollard, and I. Sag. Minimal recursion semantics: An introduction. 3(2):281–332, July 2005.
- Ann Copestake. Definitions of typed feature structures. In S. Oepen, D. Flickinger, J. Tsujii, and H. Uszkoreit, editors, *Collaborative Language Engineering*, pages 227–230. CSLI Publications, Stanford, CA, 2002.
- Joshua Crowgey. The syntactic exponence of negation: A model for the LinGO Grammar Matrix. Master’s thesis, University of Washington, 2012.
- Chris Curtis. A parametric implementation of valence-changing morphoplogy in the LinGO Grammar Matrix. Master’s thesis, University of Washington, 2018.
- Robert M.W. Dixon. *A Grammar of Boumaa Fijian*. University of Chicago Press, 1988.
- Scott Drellishak. *Widespread but not universal: Improving the typological coverage of the Grammar Matrix*. PhD thesis, University of Washington, 2009.
- Scott Drellishak and Emily M. Bender. A coordination module for a crosslinguistic grammar resource. In S. Müller, editor, *The Proceedings of the 12th International Conference on Head-Driven Phrase Structure Grammar, Department of Informatics, University of Lisbon*, pages 108–128, Stanford, 2005. CSLI Publications. URL <http://csli-publications.stanford.edu/HPSG/6/>.
- Matthew S. Dryer. Position of pronominal possessive affixes. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL <http://wals.info/chapter/57>.

- Matthew S. Dryer and Martin Haspelmath, editors. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL <http://wals.info/>.
- Dan Flickinger. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6 (1) (Special Issue on Efficient Processing with HPSG):15–28, 2000.
- Dan Flickinger. Accuracy vs. robustness in grammar engineering. In E.M. Bender and J.E. Arnold, editors, *Language from a Cognitive Perspective: Grammar, Usage, and Processing*, pages 31–50. CSLI Publications, Stanford, 2011.
- Gerald Gazdar. *Generalized phrase structure grammar*. B. Blackwell, Oxford, 1985.
- Michael Goodman. Generation of machine-readable morphological rules from human-readable input. *University of Washington Working Papers in Linguistics*, 30, 2013.
- William W. Goodwin. *A Greek Grammar*. Macmillan & Co., London, 1894.
- Pavel Grashchenkov. Tipologija posessivnyx konstrukcii (Typology of possessive constructions). *Voprosy jazykoznanija*, 55(3):25–54, 2005.
- Michael Haeger. An evidentiality library for the LinGO Grammar Matrix. Master’s thesis, University of Washington, 2017.
- John Haiman. Iconic and economic motivation. *Language*, 59:781–819, 1983.
- John Haiman. *Natural Syntax*. Cambridge University Press, Cambridge, 1985.
- Martin Haspelmath. External possession in a European areal perspective. In D. Payne and I. Barshi, editors, *External Possession*. John Benjamins, Amsterdam, 1999.
- Martin Haspelmath. Explaining alienability contrasts in adpossession constructions: Predictability vs. iconicity. *Zeitschrift fur Sprachwissenschaft*, 36(2):193–231, 2017.
- Bernd Heine. *Possession: cognitive sources, forces, and grammaticalization*. Cambridge University Press, Cambridge, 1997.

- Brian G. Hewitt. *Georgian: A structural reference grammar*. John Benjamins, Amsterdam, 1995.
- William Horton Jr. Jacobsen. *A Grammar of the Washo Language*. PhD thesis, University of California, Berkeley, 1964.
- Fred Karlsson. *Finnish: An essential grammar*. Routledge, London; New York, third edition, 2015.
- Daniel Aaron Kaufman. *The morphosyntax of Tagalog clitics: A typologically driven approach*. PhD thesis, Cornell University, 2010.
- Dodona Kiziria. *Beginner's Georgian*. Hippocrene Books, New York, 2008.
- Judith Klavans. The independence of syntax and phonology in cliticization. *Language*, 61: 95–120, 1985.
- Maria Koptjevskaja-Tamm. Adnominal possession. In W. Oesterreicher M. Haspelmath, E. König and W. Raible, editors, *Language Typology and Language Universals*, volume 2, pages 960–970. Walter de Gruyter, Berlin, 2001.
- Maria Koptjevskaja-Tamm. Adnominal possession in the European languages: form and function. *Sprachtypologie und Universalienforschung*, 55(2):141–172, 2002.
- Maria Koptjevskaja-Tamm. A woman of sin, a man of duty, and a hell of a mess: non-determiner genitives in Swedish. In *Noun Phrase Structure In The Languages Of Europe*, pages 515–558. 2003. ISBN 3-11-015748-9.
- Tibor Laczko. Revisiting possessors in Hungarian DPs: a new perspective. In M. Butt and T.H. King, editors, *Proceedings of the LFG07 Conference*. CSLI Publications, 2007.
- Steven Theophilos Lasswell. *An Ecological Reference Grammar of Sölring North Frisian*. PhD thesis, University of California Santa Barbara, 1998.

- Tom Lundskaer-Nielsen and Philip Holmes. *Danish: An Essential Grammar*. Routledge Essential Grammars. Taylor & Francis, 2011.
- Christopher Lyons. The syntax of English genitive constructions. *Journal of Linguistics*, 22 (1):123–43, 1986.
- Robert Malouf. A head-driven account of long-distance case assignment. In C. Grover R. Cann and P. Miller, editors, *Grammatical Interfaces in HPSG*, pages 201–214. CSLI Publications, Stanford, 2000.
- Donald J. Mastrorarde. *Introduction to Attic Greek*. University of California Press, 2013.
- Naomi Hanaoka McGloin. *Modern Japanese grammar: a practical guide*. Routledge, London; New York, 2014.
- Terence F. Mitchell. *Colloquial Arabic: The living language of Egypt*. English Universities Press, London, 1962.
- Charles Fessenden Morse. *A Grammar of the Bulgarian Language: With Exercises and English and Bulgarian Vocabularies*. D. Zankoff, Constantinople, 1859.
- Lea Nash. The structural source of split ergativity and ergative case in Georgian. In D. Masmam J. Coon and L.D. Travis, editors, *The Oxford Handbook of Ergativity*. Oxford University Press, 2017.
- Johanna Nichols. On alienable and inalienable possession. In W. Shipley, editor, *In honor of Mary Haas: from the Haas Festival Conference on Native American Linguistics*, pages 557–610. Mouton de Gruyter, Berlin, 1988.
- Johanna Nichols and Balthasar Bickel. Possessive classification. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL <http://wals.info/chapter/59>.

- Stephan Oepan, Dan Flickinger, Hans Uszkoreit, and Jun-ichi Tsujii. *Collaborative Grammar Engineering*. CSLI Publications, 2002.
- Kelly O’Hara. A morphotactic infrastructure for a grammar customization system. Master’s thesis, University of Washington, 2008.
- Stanley Peters and Dag Westerståhl. The semantics of possessives. *Language*, 89:713–759, 12 2013.
- Frans Plank. Possessives and the distinction between determiners and modifiers (with special reference to German). *Journal of Linguistics*, 28(2):453–468, 1992.
- Carl Pollard and Ivan A. Sag. *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. University of Chicago Press, 1994.
- Laurie Poulson. Meta-modeling of tense and aspect in a cross-linguistic grammar engineering platform. Master’s thesis, University of Washington, 2011.
- Carol Rounds. *Hungarian: an essential grammar*. Routledge, London, 2001.
- Safiyah Saleem. Argument optionality: A new library for the Grammar Matrix customization system. Master’s thesis, University of Washington, 2010.
- Gary F. Simons and Charles D. Fennig. Ethnologue: Languages of the world. <https://www.ethnologue.com>, 2018.
- Herbert Weir Smyth. *A Greek Grammar for Colleges*. American Book Company, 1920.
- Sanghoun Song. *A grammar library for information structure*. PhD thesis, University of Washington, 2014.
- Anna Szabolcsi. The noun phrase. In *The Syntactic structure of Hungarian*, volume 27 of *Syntax and semantics*. Academic Press, San Diego, 1994.

Ida Toivonen. The morphosyntax of Finnish possessives. *Natural Language and Linguistic Theory*, 18(3):579–609, 2000.

Thomas Trimble. Adjectives in the LinGO Grammar Matrix. Master’s thesis, University of Washington, 2014.

Olga Zamaraeva and Kristin Howell. Modeling clausal complementation for a grammar engineering resource. *HPSG 2018*, To be presented.