25 Construction Grammar and the Syntax-Semantics Interface

Laura A. Michaelis

Chapter Overview

1 Introduction

A foundational assumption of traditional generative grammar is that a grammar is organized in the mind of the speaker as a number of hermetically sealed modules, which in the course of a derivation hand off data one to the other.[1] Sign-Based Construction Grammar (SBCG) assumes that the grammar contains no such modules, but rather that grammar is an inventory of *signs*, complexes of linguistic constraints that contain information about form, meaning, and use, and that constructions are the means by which simpler signs are combined into more complex signs. The notion of construction, on this view, is a formalization, in a constraint-based architecture, of the notion of construction in traditional grammar. A simple illustration of a construction is the subjectless tagged sentence shown in (1):

(1) *Looks nice, doesn't it?*

In a sentence like (1) the missing subject of the main clause can only be interpreted on the basis of the reference of the tag's subject. In particular, the addressee(s) of (1) must determine what *it* refers to in order to reconstruct the missing first argument of the main clause. While idiosyncratic, the biclausal construction that licenses (1) shares properties with more general constructions, including that which licenses a question tag that is of opposite polarity to the main clause and pronounced with rising intonation, for example, *That's okay with you,* ***isn't it****?* The particular combination, arrangement, and interpretation of these inherited, construction-licensed signs is, however, peculiar to sentences of this form: a main clause missing a subject, followed by a question tag whose pronominal subject provides the reference of the missing subject of the main clause (Kay 2002). Generalizations about constructions are captured through the interaction of a hierarchical classification of types and the type-based inheritance of grammatical constraints (Sag 2010).

The approach described here is distinct from traditional generative grammar, as it is constraint-based rather than derivational and assumes neither underlying structure nor empty categories. In addition, semantic constraints and use conditions are directly associated with the phrase-structure rules that define constructions, rather than being a "read off" syntactic representation or relegated to a "pragmatic component." The constructionist program does, however, fall within generative grammar in the historically broader sense of aiming to provide a fully explicit account of the sentences of each language under study.

In this chapter, I will lay out the case for construction-based grammar, and in particular SBCG, by describing the theory's foundations, the theory's formal implementation, and facts that support a construction-based view of grammar. Section 2 will be devoted to the foundations of construction-based syntax. In Section 3, I will describe the formal foundations of SBCG. Section 4, devoted to the facts, will describe two major lines of evidence for construction-based syntax. Section 5 will provide concluding remarks.

2 Foundations

To some linguists, Construction Grammar seems to make an obvious point. Why would anyone think that syntax *isn't* based on constructions? After all, the category of construction has been a part of grammatical discourse since ancient times. But while we find continuations of that tradition in pedagogical and field grammars, as far as syntacticians are concerned, construction-based analysis stopped making theoretical sense when grammar was redefined, as per transformationalist assumptions, as a mechanism for assembling symbols into phrases. According to the transformationalist tradition, syntactic rules do only one thing:

determine what symbol sequences function as units for syntactic purposes. They cannot add conceptual content to that contributed by the words. If sentence meaning does not come from "construction meaning," there seems little point in positing constructions. In fact, in the "rule-free" conception of grammar promoted by Chomsky (1989, 1995), grammatical constructions are "taxonomic epiphenomena" whose properties are predictable from the interaction of fixed principles with language-particular parameter settings (Chomsky 1989: 43).

The difference between a construction-based approach to grammar and one based on interacting universal principles can be viewed in part as a distinction between a positive licensing strategy—ruling certain structures in—and a negative suppression-based strategy—ruling certain structures out (Zwicky 1994; Malouf 2003). The constraints in a suppression-based theory like Government and Binding (GB) theory include the case filter, the binding principles and restrictions on long-distance dependencies, for example, subjacency. The theory is deemed successful if each of the ill-formed sentences of the language under study violates at least one constraint. By contrast, according to the licensing-based view of grammar adopted by SBCG

> [a]n expression is syntactically well-formed if its phonological form is paired with its semantics as an instance of some syntactic construction. It follows that an expression is ungrammatical only because there is no combination of constructions that license it, not because there is some cross-constructional filter that rules it out. (Zwicky 1994: 614)

As observed in the above passage, constructions interact in the licensing of language objects. That is, a linguistic expression can instantiate multiple types at once. For example, the fronting pattern exemplified by (2) is an instance of both the Topicalization construction and the Filler-Gap construction:

(2) *That I'm not so sure about.*

These kinds of interactions are described by type hierarchies, as discussed by Sag (2010).

Why would anyone prefer a licensing-based model? After all, suppression-based syntactic theories offer constraints of potentially universal significance, and they are inarguably more economical than licensing-based models, since there are far fewer general constraints than there are constructions. Construction grammarians prefer the licensing model not because it is more elegant, but because it provides descriptive precision that suppression-based approaches cannot.

Construction Grammar retains descriptive goals that generative-transformational grammar long ago exchanged for the promise of bright-line tests that would separate the relevant ("core") grammatical phenomena from the irrelevant ("peripheral") ones. If one takes Chomsky's claims seriously, the loss of descriptive coverage that resulted from this move was a sign of progress in the "search for explanatory adequacy" (Chomsky 1995: 435). But, as Sag (2010), observes, the generative-transformational tradition finesses the core phenomena too: when one considers that tradition's signature phenomenon, the English filler-gap dependency, one finds that it is silent concerning salient parameters of variation among the extraction constructions, including the syntactic categories of the filler and head daughters, the type of *wh*-element in the filler daughter (if any), and the use of the auxiliary-initial pattern in the head daughter. This in turn should lead us to ask how a theory that takes *cross-linguistic* parametric variation seriously can overlook *intra-linguistic* variation of a similar nature.

3 Formalism

For many years, the only formal reference work available to construction grammarians was an unpublished (but widely circulated) course reader, Fillmore and Kay (1995). It outlines a model that has come to be known as Berkeley Construction Grammar (BCG). This work contains a compelling treatment of interactions between argument-structure constructions (e.g. passive and ditransitive) and shows that one can use the mechanism of lexeme-construction unification to describe English nominal and verbal syntax without recourse to the unary-branching phrases and "inaudible" determiners of X′-based approaches. However, this work also uses a cumbersome nested-box notation for constructions that permits an undefined degree of recursion, and an open-ended and loosely organized repertoire of features. In addition, while Fillmore and Kay (1995) persuasively argue that formal and semantic commonalities among constructions can be captured by means of inheritance relations (rather than, say, transformations), the work does not provide any means of notating such taxonomic relationships other than scattershot notations in construction diagrams. Construction grammarians seeking a more comprehensive and principled system of formal representation were inclined to look to an allied declarative model, Head-Driven Phrase-Structure Grammar (HPSG; Pollard and Sag 1987; 1994). Like BCG, HPSG treats words and phrasal patterns as similar kinds of form-meaning pairings, uses feature structures to model semantic and syntactic classes of grammar objects, and assumes a hierarchical classification of grammatical structures.

SBCG, described by Sag (2010, forthcoming), Kay and Sag (forthcoming), and Michaelis (2009), among others, is an attempt to "expand the empirical coverage of HPSG, while at the same time putting BCG on a firmer theoretical footing" (Sag forthcoming). SBCG is a theory of constructional meaning because

9781441124609_Ch25_Fpp_txt_prf.indd 424 6/12/2012 5:40:10 PM

it assumes that rules of syntactic combination (descriptions of local trees) are directly associated with interpretive and use conditions, expressed by semantic and pragmatic features that attach to the mother or daughter nodes in these descriptions (Kay and Michaelis forthcoming, Sag forthcoming). This amounts to the claim that syntactic rules have meanings. This claim sets Construction Grammar apart from prevailing models of meaning composition. Such theories are based on a principle that Jackendoff (1997: 48) describes as the "doctrine of syntactically transparent composition." According to this doctrine, "[a]ll elements of content in the meaning of a sentence are found in the lexical conceptual structures [. . .] of the lexical items composing the sentence" and "pragmatics plays no role in determining how [lexical conceptual structures] are combined." To propose a construction-based model of semantic composition like SBCG is not, however, to deny the existence of syntactically transparent composition. It is instead to treat it, in accordance with Jackendoff (1997: 49), as a "default in a wider array of options." That is, whenever a class of expressions can be viewed as licensed by a context-free phrase-structure rule accompanied by a rule composing the semantics of the mother from the semantics of the daughter, a construction-based approach would propose a construction that is functionally equivalent to such a rule-to-rule pair. But the constructional approach also enables us represent linguistic structures in which the semantics of the mother does not follow entirely from the semantics of the daughters. In this section, we will discuss two features that make SBCG a useful formalism for construction-based syntax. The features are locality and variable-grain description.

3.1 Locality

In SBCG, the phrase types in the target language are described by means of combinatory constructions. Combinatory constructions describe constructs—signs that are built from one or more distinct signs. Constructions in SBCG take the form of type constraints. A type constraint is a conditional statement that tells what properties a construct will have if it is an instance of the type in question. Intuitively, constructs are local trees (mother–daughter configurations) with feature structures (specifically, signs) at the nodes. Constructions can describe only such mother–daughter dependencies and not, for example, mother–granddaughter dependencies (Sag 2007, 2010).

A construct is modeled in SBCG as a feature structure that contains a mother (MTR) feature and a daughter (DTRS) feature. The value of the MTR feature is a sign and the value of the DTRS feature a list of one or more signs. What then is a sign? A sign, as in the Saussurean tradition, is a form–meaning pairing. A sign is modeled as a type of feature structure, or attribute–value matrix. A sign specifies values for six features (Sag 2010, forthcoming):

- PHON
- FORM: a list of formatives comprising the expression
- ARG-ST: a ranked list of a lexical expression's arguments
- SYN: CAT and VAL(ENCE)
- SEM: INDEX and FRAMES
- CNTXT: TOPIC and FOCUS

By treating phrases as feature structures, SBCG captures properties common to lexemes and phrase types in a way that BCG did not. As mentioned, according to the BCG vision, the grammar is an inventory of trees (nested boxes) with an indefinite depth of recursion. By contrast, argument-structure constructions like the Transitive construction are represented by feature structures, as in Figure 25.1.

The construction shown in Figure 25.1 expresses a constraint on transitive lexemes: each such lexeme assigns the grammatical relation (*rel*) *object* to one argument in its valence (*val)* set, provided that this argument is not the highest ranking or "distinguished" argument (hence the negative value assigned to the feature *DA*, or "distinguished argument"). The Transitive construction presumably represents a class of lexemes (those that take direct objects), but it is unclear why a lexeme description like that in Figure 25.1 should qualify as a construction, as it does not contain nested boxes. SBCG, by contrast, proposes two types of constructions: the aforementioned combinatory constructions, which describe properties of phrase types, and lexical-class constructions, which in turn describe properties shared by classes of lexemes (like *devour*) and words (like *devoured*). The only difference between lexical-class constructions and combinatory constructions is the type name in the antecedent of the type constraint. Because both words and phrases are signs, the two can be described

$$\begin{bmatrix} \text{syn [voice active]} \\ \text{val} \left\{ \begin{bmatrix} \text{rel} \begin{bmatrix} \text{gf obj} \\ \text{DA} - \end{bmatrix} \end{bmatrix} \right\} \end{bmatrix}$$

Figure 25.1 The Transitive construction as per BCG (Fillmore and Kay 1995)

$$\textit{applicative lexeme} \Rightarrow \begin{bmatrix} \textit{trans - verb - lexeme} \\ \\ \text{ARG -ST} < \text{NP}x, \text{NP}z, \text{PP}[\textit{with}]y > \\ \text{SEM} \mid \text{FRAME} \begin{bmatrix} \textit{saturation - frame} \\ \text{ACTOR } x \\ \text{THEME } y \\ \text{SURFACE } z \end{bmatrix} \end{bmatrix}$$

Figure 25.2 The Applicative lexical-class construction

$$subject\text{-}pred\text{-}cxt \Rightarrow \begin{bmatrix} \textit{phrase} \\ \text{MTR} \begin{bmatrix} \text{SYN [VAL} < \ > \\ \text{MRKG } M : \textit{unmk} \end{bmatrix} \\ \text{DTRS } < X, H > \\ \text{HD-DTR} : H \begin{bmatrix} \begin{bmatrix} \text{CAT [VF } \textit{fin}] \\ \text{VAL} < X > \\ \text{MRKG } M : \textit{unmk} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 25.3 The Subject-Predicate combinatory construction

in the same way. This is shown by Figures 25.2–25.3, which illustrate, respectively, a lexical-class construction and a combinatory construction.

The Applicative construction, shown in Figure 25.2, describes the lexeme class to which the verbs *fill* and *cover* belong (as in, for example, *She filled the bathtub with champagne* and *They covered the wall with a sheet*): this lexeme class is a subtype of the transitive-lexeme (*trans-verb-lexeme*) class, as indicated by the typing of the feature structure to the right of the arrow. As shown by the ARG-ST list, verbs of this lexeme class express the theme argument as a PP headed by *with*. The semantic constraints associated with this lexeme class are, as indicated by the frame, labeled *saturation-frames* in the FRAMES list. This frame is intended to capture the resultant-state entailment that the theme occupies a critical mass of points in a planar region (Michaelis and Ruppenhofer 2001). The Applicative construction also describes one of the classes to which the verbs *spray* and *load* belong: the lexical entries of these verbs lack an ARG-ST list, making them compatible with both the Applicative and Oblique-Goal lexical-class constructions.

The Subject-Predicate construction, shown in Figure 25.3, licenses basic declarative clauses. This construction contains two features not previously discussed: HD-DTR and MARKING (MRKG). The value of the HD-DTR feature is a syntactic feature structure that describes the head sign in a phrase. The INDEX value of this feature structure is identical to that of the second sign in the DTRS list. The value of the MRKG feature is the type of the grammatical marker that accompanies a sign (e.g. *that* in the case of phrasal verbal sign whose DTRS are the complementizer *that* and a clause). In the case of the Subject–Predicate construction, the value *unmk* indicates that neither the verbal head nor its MTR has a grammatical marker. As described in Figure 25.3, a subject–predicate construct consists of two daughter signs, the second of which is a finite verbal sign that selects for the first sign by means of its VAL feature. As shown in this figure, the MTR of a subject–predicate construct has an empty VAL set, indicating that it is a complete predication.

Thus, SBCG captures properties common to lexical items and phrases by describing both as feature structures. It is true that combinatory constructions describe sign configurations (via the MTR and DTRS features), while lexical-class constructions describe single signs. But signs and sign configurations are the same thing as far as the licensing mechanism is concerned. The principle that governs the licensing of language objects in SBCG is the Sign Principle. According to the Sign Principle of SBCG (Sag forthcoming), a sign is *lexically licensed* if it satisfies a lexical entry and *constructionally licensed* if it is the mother sign of some construct type.[2] This means that one can verify the grammaticality of a phrase based only on the properties of its topmost (MTR) feature structure, since these properties include identifying information about that node's daughters (e.g. the frames on the MTR's FRAMES list).

3.2 Variable Granularity

As is widely recognized by proponents of Construction Grammar and exemplar-based approaches (e.g. Bybee 2007), many grammatical generalizations are not very general. Thus, grammar must contain constraints of varying grains. Independent-clause (IC) exclamatives (e.g. *What fools they are!*) provide an example of a fine-grained constraint (Sag forthcoming):

(3) *God, *(I can't believe) who they hired/where they went!*

As (3) shows, IC and subordinate-clause exclamatives differ with regard to the syntactic category of the filler daughter: *who* and *where* are not possible filler daughters of IC exclamatives in English, although they are in some other languages (Michaelis 2001).

A grammar that provides no mechanism for imposing category restrictions will overgenerate. How does SBCG avoid overgeneration? It treats nodes, and in particular the MTR nodes of constructs, as feature structures—not category labels. A description of a feature structure is a set of properties. As property sets, feature-structure descriptions follow the logic of set inclusion: the more properties in the description, the smaller the class of language objects that description picks out. For example, the feature set that describes an IC exclamative includes that which defines the filler-head construction, shown in Figure 25.4. Inclusion

$$\textit{filler-hd cxt} \Rightarrow \begin{bmatrix} \text{MTR} \begin{bmatrix} \text{SYN [VAL } L1] \\ \text{GAP } L2 \end{bmatrix} & \\ \text{DTR } < [\text{SYN } X], H > & \\ \text{HD-DTR } H\text{:} & \begin{bmatrix} \textit{Phrase} \\ \text{SYN} \begin{bmatrix} \text{CAT } \textit{verbal} \\ \text{VAL } L1 \end{bmatrix} \\ \text{GAP } < [\text{SYN } X] > \end{bmatrix} \end{bmatrix}$$

Figure 25.4 The Filler-Head construction relations among feature-structure descriptions allow us to model constructs at each step along the idiomaticity continuum, with an array of constructions of correspondingly graded generality.

4 Facts

The descriptive goals of modern syntacticians are broad, and concern general phenomena like the discourse-syntax interface, meaning composition and argument realization. Thus, the challenge for proponents of construction-based syntax is to demonstrate that constructions figure in the most basic functions that grammar performs. In this section, I will discuss ways to confront this challenge, highlighting two fundamental grammar functions: semantic composition and complement licensing. In Section 4.1, I will argue that a sensible model of semantic composition requires recourse to constructional meaning. In Section 4.2, I will give evidence that constructions are licensors of complements.

4.1 Composition Requires Constructions

Let us assume, following the tradition of Generalized Phrase Structure Grammar (Gazdar et al. 1985), that for some class of expressions it is licensed by a phrase-structure rule and that this phrase-structure rule is paired with an interpretive rule that composes the semantics of the mother from the semantics of the daughters. In such cases, Construction Grammar would propose a construction that does the same work that such a rule-to-rule pair does. Obviously, we have not validated a constructionist approach by simply showing that it replicates what other theories do without constructions. But Construction Grammar can also describe linguistic structures in which the mother of a given local tree may ~~have more than one interpretation~~. Can a syntactic theory based on strict composition do the same thing? It appears that the answer is no, at least if we use the following definition of compositionality, taken from the Szabó (2007): "If a language is compositional, it cannot contain a pair of non-synonymous complex expressions with identical structure and pairwise synonymous constituents." The problem with this understanding of meaning composition is that it yields a counterintuitive result for syntactically regular idioms of the type described by Fillmore et al. (1988). Two examples of such idioms are given in (4) and (5), respectively:

(4) Pseudoimperative*: Now watch me be somehow unable to make it out there.*

(5) Pseudoconditional*: If you're Obama, you might not like the idea of the Clintons in the White House.*

Both the Pseudoimperative and the Pseudoconditional are syntactically regular: the former has a syntactic form indistinguishable from that of a regular imperative sentence and the latter has the syntactic form of an ordinary hypothetical conditional. But (4) does not exhort the hearer to watch the speaker and (5) does not predict a future outcome based on the hearer's ability to shift identity. If we are to maintain Szabó's definition of compositionality, we must conclude from these interpretive affordances either that English is not compositional or that Pseudoimperatives and Pseudoconditionals have distinct hierarchical representations from their transparently interpreted analogs. Neither appears to be an acceptable conclusion. Thus, a conception of composition based exclusively on X′-syntax leads to a nonsensical result for examples like (4–5). By contrast, a constructional approach allows a single phrase-structure analysis for both the idiomatic and transparent readings of (~~7–8~~) and, for the idiomatic readings, posits constructions that attach semantic interpretations directly to complex syntactic objects. For example, under a constructionist analysis, the Pseudoconditional and the regular conditional have a common supertype that is used to define the properties that the two subtypes have in common. In short, constructional approaches recognize as instances of compositionality cases in which two different meanings for the same syntactic form are licensed by two different collections of form-meaning licensers, that is, by two different collections of constructions.

4.2 Constructions License Complements

Here we will consider two lines of evidence supporting the contention that constructions license complements: valence variation and "weird sisterhood."

4.2.1 Valence Variation

According to Rappaport Hovav and Levin (1998), henceforth RHL, valence augmentation and other contextual effects on verb meaning are the products of lexical derivations that build up complex event structures from simpler ones. Unlike the construction-based model of argument structure proposed by Goldberg (1995), the RHL model is based on lexical projection; as they put it: "Many aspects of the syntactic structure of a sentence—in particular, the syntactic realization of arguments—are projected from the lexical properties of the verbs" (RHL: 97). Each of a verb's syntactic frames is associated with a distinct verb meaning, although every verb has one basic class membership. The more aspectual representations a verb has the more syntactic variation it will display, and vice versa. To represent verb meaning and semantic operations on verb meaning, RHL propose (i) a set of Aktionsart-based templates and (ii) an operation that augments one such schema up to another one. The

RHL model assumes that verbs unify with event-structure templates based on Aktionsart class. According to this model, activity verbs like *sweep* are lexically intransitive, although such verbs can gain a second argument by combining with an accomplishment template, as in, for example, *She swept the floor*. Arguments supplied by templates are referred to as *nonstructural arguments*.

The RHL model makes three predictions about null complements (Ruppenhofer 2004; Goldberg 2005, 2006):

- *As nonstructural arguments, the second arguments of bivalent state, achievement and activity verbs should always be omissible, for example, Have you eaten?*
- *Nonstructural participants are subject only to a recoverability condition based on prototypicality (p. 115); therefore all null complements should have existential (indefinite) interpretations, as in, for example, She reads.*
- *As structural arguments, patient arguments of accomplishment verbs, for example, kill and break, should never be omissible.*

However, each of these predictions proves false. First, the second argument of a bivalent state, achievement, or activity verb is not always omissible, as shown by (6–8), respectively:

(6) *She enjoys *(things).*
(7) *She found *(something).*
(8) *We discussed *(issues).*

Second, null instantiated second arguments need not have indefinite interpretations; they may instead be interpreted as zero anaphors, as in (9)–(11):

(9) *I remember (that).*
(10) *I prepared (for that event) for weeks.*
(11) *She arrived (there).*

Third, as shown by Goldberg (2005), accomplishment verbs do allow null instantiated patient arguments. For example, verbs of emission and ingestion license existential null complements, as in (12)–(13):

(12) *He cried (lachrymal fluid) into his beer.*
(13) *He swallowed (saliva) nervously.*

In addition, as observed by Ruppenhofer (2004), almost any accomplishment verb in an iterated-event context allows an existential null complement, as in (14)–(16):

(14) *The police only arrest if there is a high-profile situation.*
(15) *You just take and take.*
(16) *She has never failed to impress.*

We will now ~~look at~~ how a construction-based model of null complementation, proposed by Kay (2004), circumvents the problems described above. Instead of defining a class of "structurally intransitive" verbs, the constructionist model posits a *null-complementation construction*. This construction licenses a lexeme whose ARG-ST list contains one more sign than its VAL set does. That is, one of the arguments on the ARG-ST list lacks a corresponding valence member. Among the frames in the FRAMES set of this lexeme is a quantifier frame, which represents the construal (existential or anaphoric) of the missing valence member.

The null-complementation construction is a type of derivational construction. A derivational construction builds a new lexeme from one or more lexical signs: the mother of a derivational construct is of the type *lexeme*; its daughters are lexemes. The null-complementation construction builds a lexeme with a covert valence member from a lexeme with an optionally covert valence member. A construct licensed by the null-complementation construction is shown in Figure 25.5.

$$
\begin{bmatrix}
\text{MTR} & \begin{bmatrix}
\textit{lexeme} \\
\text{FORM} \; < \textit{eat} > \\
\text{ARG-ST} \; \langle \text{NP}_i, \text{NP}_x \rangle \\
\text{SYN} \mid \text{VAL} \; \left\langle \text{NP} \begin{bmatrix} \textit{overt} \\ \text{INST i} \end{bmatrix} \right\rangle \\
\text{SEM} \mid \text{FRAMES} \; \left\langle \begin{bmatrix} \textit{eat}-\textit{fr} \\ \text{EATER } i \\ \text{FOOD x} \end{bmatrix}, \begin{bmatrix} \textit{timed}-\textit{meal}-\textit{fr} \\ \text{INST } x \end{bmatrix}, \begin{bmatrix} \textit{exist}-\textit{fr} \\ \text{BV } x \end{bmatrix} \right\rangle
\end{bmatrix} \\
\text{DTRS} & \left\langle \begin{bmatrix}
\textit{lexeme} \\
\text{FORM} \; < \textit{eat} > \\
\text{ARG-ST} \langle \text{NP}_i, \text{NP}_x \rangle \\
\text{SYN} \mid \text{VAL} \; \left\langle \text{NP} \begin{bmatrix} \textit{overt} \\ \text{INST } i \end{bmatrix}, \text{NP} \begin{bmatrix} (\textit{ini}) \\ \text{INST x} \end{bmatrix} \right\rangle \\
\text{SEM} \mid \text{FRAMES} \; \left\langle \begin{bmatrix} \textit{eat}-\textit{fr} \\ \text{EATER } i \\ \text{FOOD } x \end{bmatrix} \right\rangle
\end{bmatrix} \right\rangle
\end{bmatrix}
$$

Figure 25.5 A null-complementation construct

In the construct shown in Figure 25.5, the DTR lexeme is the verb *eat*, which licenses two arguments, an eater and a food. The sign types of its valence members are, respectively, *overt* and *(ini)*. According to the hierarchy of sign types laid out by Kay (2004), this *(ini)* sign type may resolve to either an *overt* sign or an *ini* sign. In the construct shown in Figure 25.5, it has resolved to *ini*. The reason that we do not see this *ini* sign in the valence set of the MTR lexeme is that an *ini* sign is a type of *covert* sign; as such, it is subject to the constraint in (17):

(17) *covert* ⇒ *sign & [FORM < >]*

The constraint in (17) ensures that the type *ini* has no form value. A sign with no form value will not appear on a lexeme's VAL list, since the VAL value is a list of *overt* signs. As a subtype of the type *null-comp*, the *ini* sign type obeys the constraint in (18):

(18) $\textit{null-comp} \Rightarrow \textit{covert}\ \&\ \left[\text{SEM}\begin{bmatrix}\text{INDEX}\ x \\ \text{FRAMES}\left\langle\begin{bmatrix}\textit{frame} \\ \text{BV}\ x\end{bmatrix}\right\rangle\end{bmatrix}\right]$

The constraint in (18) ensures that if there is a sign of the type *null-comp*, the lexeme that licenses this sign will have a quantifier frame on its FRAMES list. This quantifier frame takes a bound variable (BV) as its argument, and this bound variable shares its index with the covert sign. The constraint in (19) ensures that the quantifier frame is in particular an existential-quantifier frame:

(19) *ini* ⇒ *null-comp & [SEM [FRAMES <exist-fr>]]*

The above constraint captures the existential interpretation of a missing argument in sentences like (20):

(20) *I've eaten.*

Sentence (20) means something like "I've eaten some food at a canonical meal time" rather than "I've eaten *that* food at a canonical meal time."

Described in procedural terms, the null-complementation construction that licenses the construct in Figure 25.5 builds an *eat* lexeme in which the food argument is missing from the verb's VAL set but remains part of its FRAMES set, where it is the bound variable of an existential quantifier. It is important to note, however, that while the mother and daughter lexemes have distinct VAL sets, they have the same ARG-ST set: the food participant appears on the ARG-ST list of the MTR lexeme even though it does not appear on the MTR lexeme's VAL list.

Additional null-complementation constructions are required to account for the fact that null-complementation restrictions on verbs can be overridden in certain contexts. As mentioned ~~in (8)~~ above, patient arguments of accomplishment verbs are not generally subject to null instantiation. For example, sentences like (21)–(23) sound awkward:

(21) *The police arrested *(someone) last night.*
(22) *Sam took *(something) without permission.*
(23) *I just impressed *(someone)!*

However, as observed by Goldberg (2005), accomplishment verbs that do not allow unexpressed patient arguments in episodic contexts invariably allow them in habitual-generic predications and existential-perfect predications, as shown by (24)–(26), repeated from (14)–(16) above:

(24) *The police only arrest if there is a high-profile situation.*
(25) *You just take and take.*
(26) *She has never failed to impress.*

The contexts illustrated here map neatly to constructions—the habitual present-tense inflectional construction in (24) and (25) and the perfect-participle derivational construction in (26). Let us presume that the MTR lexemes of these constructions resemble the lexeme daughter *eat* in the derivational construct shown in Figure 25.5, in that its second valence member has the type *(ini)*. This means that constructions like the habitual-present tense inflectional construction can "feed" the indefinite null-instantiation construction, thus licensing null-object *arrest* and null-object *take* in (24)–(25), respectively.

4.2.2 Weird Sisterhood

Many verb frames specify sisterhood relations that are not predicted by the general-purpose constituency rules that combine heads and complements and heads and specifiers. Many of these patterns have specialized communicative functions. A look at these phenomena suggests that highly detailed constructions, rather than non-category-specific phrase-structure rules, pair predicates and their complements. In this section, we will look at just one case of weird sisterhood: Nominal Extraposition.

In Nominal Extraposition (Michaelis and Lambrecht 1996), an exclamatory adjective, for example, *amazing*, licenses an NP complement:

(27) *I know it's just it's unbelievable the different things that are happening in America today.*
(28) *I'll date myself a little bit but it it's remarkable the number of those things they need.*
(29) *I know. I love that game. It's amazing the words they come up with.*

The pattern exemplified in (27)–(29) is idiosyncratic in two respects. First, adjectives are not case assigners and should not therefore license non-oblique NP complements. Second, this NP complement is interpreted as denoting a scalar degree (Michaelis and Lambrecht 1996). In (29), for example, the NP *the words they come up* stands in for a scalar expression like "the number of words they come up with"; that is, it is not the words themselves but their numerousness that is deemed remarkable. The fact that the complement of *amazing* in (29) has a scalar interpretation follows from the fact that (29) is an exclamation, but the pairing of an exclamatory adjective with an NP sister that denotes a degree, metonymically or otherwise, requires a construction that provides for this syntax and this meaning.

5 Conclusion

What makes construction-based syntax a radical departure from formal syntactic theory as we know it? While transformational-generative syntacticians have sought to import syntactic generalizations into the lexicon (as illustrated by the approaches of Baker 1996; Hale and Keyser 1998; and Borer 2001), construction grammarians have moved in the opposite direction: it is the lexicon that provides a model for the syntax-semantics interface in Construction Grammar. Lexical-class constructions, which define classes of lexemes or words, and combinatory constructions, which define classes of phrases, are both constraints on feature structures. In phrasal constructions, a list-valued feature of the mother is used to represent the property of having the daughters it does. Thus, rather than seeing syntax, semantics, and lexicon as independent modules, with the lexicon characterized as a bag of idiosyncratic form-meaning pairings, SBCG proposes a lexicon structured by hierarchically organized lexical classes and extends this model to relations among classes of phrases.

Notes

1. Portions of this chapter appear as Michaelis *forthcoming*.
2. According to the Sign Principle, a *lexical* sign can be *constructionally* licensed, if it corresponds to the MTR sign of a derivational or inflectional construction. In fact, the only lexical signs that are licensed by lexical entries are those that are not "produced" by derivational or inflectional constructions.