An Implementation of Filler-Gap Constructions Using the Tree-Adjoining Grammar Formalism

by

Roshan Pulapura

presented to the Department of Linguistics

and the Department of Computer Science

in partial fulfillment of the requirements

for the degree with honors

of Bachelor of Arts

Harvard College

April 2020

Contents

0 Abstract

The objective of this thesis is to evaluate a formal Construction Grammar approach to modeling filler-gap structures in comparison to the standard wh-movement procedure. Although developing formalisms for theories of grammar has been a primary focus of the field of Computational Linguistics in recent years, theories such as Construction Grammar which defy basic Chomskyan principles of Universal Grammar are comparatively under-studied. Filler-gap sentences have been previously identified as an area of syntax on which Construction Grammar may perform well, and which creates several problems for the standard approach of wh-movement. Therefore, this paper will develop an implementation of the basic principles of Construction Grammar using the popular Tree-Adjoining Grammar formalism. Using this implementation, it will define four major types of filler-gap constructions and evaluate how successful these constructions are in avoiding ungrammatical forms produced by wh-movement. Ultimately, it is determined that the Construction Grammar approach does indeed alleviate the issues of generalization and over-generation found in some sentences formed via wh-movement. However, several corresponding limitations to the Construction Grammar approach are also identified, including under-generation and unwieldy specificity. Since not enough information exists outside of this small case study to support the value of Construction Grammar in comparison to more widely accepted theories, this paper instead concludes that the formal Construction Grammar approach at least merits further study, particularly through practical experimentation.

1 Introduction

In recent years, the field of computational linguistics has increasingly been concerned with investigating the viability of various mathematical formalisms for rigorously defining the grammar of a natural language. These formalisms have ranged from Noam Chomsky's Context-Free Grammars (CFG) developed throughout the 1950's and subsequent Linear Context-Free Rewriting systems, to Lecomte and Retorte's 2001 Minimalist grammars, to Aravind Joshi's increasingly popular tree-rewriting formalism Tree-Adjoining Grammar, which will form the basis of this research. However, the majority of mathematical formalisms applied to the modeling of natural language have accepted some variant of Chomsky's theory of Universal Grammar (UG)—that is, while morphemes and individual lexical items are acquired by children through exposure to adult speakers, the range of possible syntactic and semantic structures of a language are innate knowledge of all human beings. The process of learning grammar natively, then, is simply a matter of identifying via exposure to fluent speakers which syntactic and semantic features are used in a given language and which are not. A wide range of formalisms have been developed including phrase-structure grammars such as Sag and Pollard's Head-Driven Phrase Structure Grammar (HDPSG) and Chomsky's Minimalist Program (MP), and dependency grammars such as Lexicase and Operator Grammar. These formalisms have been implemented programmatically for use in the field of Artificial Intelligence, particularly in the realm of Natural Language Processing and Human-Computer interaction.

Far less research has been done towards the development of such formalisms for alternative theories of grammar which reject the central hypothesis of UG. Although UG and its variants are largely accepted in the field of theoretical linguistics, and extensive psychological and neuroscientific data exists to support it, several other theories of grammar have been proposed and studied in an attempt to explain some of the perceived failings of UG. One such theory is Construction Grammar (CxG), which will be the subject of this research. It is the author's belief that alternative theories should be investigated, formalized, and empirically compared to models developed from UG, which has several well-documented inconsistencies and weaknesses, some of which are enumerated in the following chapter. Although at this time there is an absence of concrete evidence to suggest that CxG is more "correct" than UG in terms of how accurately it describes the actual biological processes involved in human language acquisition, it is certainly worth investigating whether or not CxG based models offer any improvement over UG models in areas such as machine comprehension or text generation.

Since a complete formalism of Construction Grammar for the English language is far beyond the scope of this thesis, it will instead focus on developing a formalism for what Ivan Sag (2010: 489-495) has referred to as "filler-gap constructions," which are covered by wh-movement in Chomsky-based theories. These constructions are characterized by a "filler" clause that occurs at the beginning of the construction and a main clause with some unfilled complement or adjunct, i.e. the "gap." The four filler-gap constructions in question are topicalized clauses, wh-exclamatives, wh-questions, and wh-relatives.

(1.1) **Waffles**, I like [] (but pancakes, I don't).

(1.2) **What a good movie** that was []!

(1.3) **Whose father** did you see []?

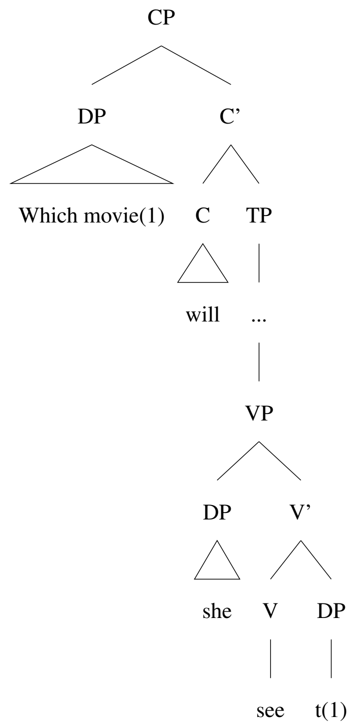
(1.4) I liked the book **which** she brought [] to school.

In the above examples, the filler clause is boldfaced, and the gap in the main clause is bracketed. Although Sag identifies several other filler-gap constructions, this project will focus on these four most commonly occurring examples.

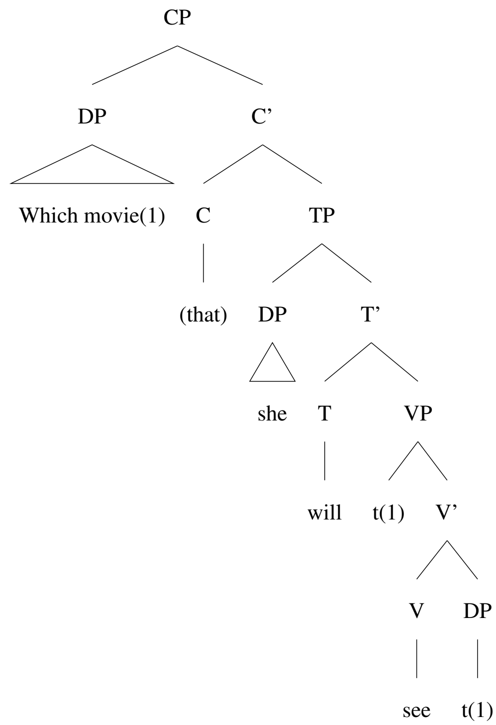
Filler-gap constructions in particular are useful to study from a Construction Grammar approach because they highlight some of the primary failings of UG, which are overgeneration of ungrammatical sentences due to imprecise phrase-structure rules and generalization of syntactic structures with distinct semantic interpretations. Sag notes that topicalized clauses, wh-interrogatives, wh-relatives, and wh-exclamatives are all accounted for using the wh-movement operation in mainstream generative grammar, although the various filler-gap structures differ both syntactically and semantically from one another. Syntactically, they diverge on which wh- elements are permissible in the filler, which types of clauses are permissible as the main clause, the possible syntactic categories of the daughter of the filler, the invertibility of the daughter, and so forth. The four constructions also diverge on the semantic categories to which they belong: propositions, questions, and exclamations (Sag 2010: 490). Therefore, while the move operation can produce a unique deep structure from given grammatical speech, it is insufficient to precisely identify whether or not a given surface structure is grammatical, without extensive further constraints. This is, of course, particularly problematic from a computational perspective. An artificial intelligence should not only be able to produce and interpret grammatical speech, it should also fail to produce and interpret ungrammatical speech.

Figures 5-6 show examples of how using the sparse rules of wh-movement with no further restrictions would cause "over-generation" of nongrammatical filler-gap sentences. Following the rules of wh-movement in Figure 5, the grammatical surface structure "Which movie will she see?" is translated back to the deep structure with a trace at the complement of "see" so that "movie" may be interpreted as the object of "see." However, in Figure 6, following the rules of wh-movement, the ungrammatical form "Which movie she will see?" may also be translated back to a deep structure with a trace at the complement of "see" following the same rules of wh-movement. But the sentence "I wonder which movie she will see?" is once again grammatical following the rules of wh-movement without any changes to the content of the filler-gap segment illustrated in Figure 6. It is evident that further constraints on the word order of the gap clause and the surrounding context are necessary in addition to the simple, generalized move operation.

(1.5)



(1.6)



Having established the weaknesses of the wh-movement approach to filler-gap sentences, the objective of this paper, then, is to determine whether a Construction Grammar approach might resolve these issues of generalization and over-generation. The central thesis of Construction Grammar is that the lexicon of any language consists of a continuum of individual vocabulary words and syntactic and semantic structures. A Construction Grammar approach, therefore, must treat syntactic structures as lexemes which are acquired in childhood, in contrast to the UG approach which posits that children have an innate comprehension of cross-linguistic syntactic structures and instead learn which ones are appropriate for their native language. A thorough description of the principles of Construction Grammar and a definition of "grammatical construction" are presented in the following chapter. In order to assess the viability of Construction Grammar in comparison to UG on the issue of filler-gap sentences, a formalism of filler-gap constructions has been presented using Tree-Adjoining Grammar (TAG). TAG is a mathematical formalism which has been extensively used to model linguistic theories and was selected due to its success in other linguistic research. A description of the TAG formalism will be presented in Chapter 2. Chapter 3 will present an implementation of the principles of Construction Grammar in the TAG formalism. The category of "grammatical constructions" is defined in terms of the finite set of elementary trees for a given TAG. Elementary trees (i.e. constructions) are designated as initial or auxiliary depending on whether or not they can be used to recursively produce language. The two fundamental operations of TAG, adjunction and substitution, are used to model the compositional and inheritance relationships between grammatical constructions. Chapter 4 will use the TAG implementation of Construction Grammar to specifically define all the constructions necessary to produce and interpret the four categories of filler-gap sentences outlined above. Finally, Chapter 5 will assess how well the Construction Grammar approach was able to resolve the issues presented with the wh-movement approach. In most of the well-documented cases of wh-movement overgenerating or generalizing, the Construction Grammar approach is able to restrict the syntactic categories of the filler, main clause, and gap sufficiently and presents an improvement over the popular UG approach by this means of assessment. However, the Construction Grammar approach introduces the corresponding problems of under-generation and extreme specificity. Chapter 5 ultimately argues that the Construction Grammar approach to filler-gaps is worthy of further study especially from a computational perspective because while it greatly expands the base of primary linguistic structures a human or machine must acquire, it significantly simplifies the number of rules and restrictions required to combine these basic structures grammatically. This study concludes with a presentation of potential future avenues of research in terms of improving the formalism already presented, introducing more rigorous means of evaluating it in comparison to pre-existing models, and applying it in an engineering context to the development of better automated speech production and parsing.

2 Background

2.1 Introduction

The goal of this chapter is to define the linguistic theories and mathematical tools which are necessary to develop a TAG formalism for filer-gap constructions. It will begin with a description of the fundamental principles of Construction Grammar (CxG) and demonstrate, through prior research, why those principles make CxG a viable alternative to UG. It will use this information to explain why filler-gap constructions are particularly useful in demonstrating the potential of CxG over UG, because the principles of CxG can address the issues of syntactic overgeneration and semantic generalization present in UG descriptions of the same phenomena. It will then address the only notable existing mathematical formalism for CxG, Fluid Construction Grammar (Steels 2011), and note the advantages and disadvantages of this formalism. This chapter will also introduce Tree-Adjoining Grammar, developed by computer scientist Aravind Joshi, as well as a set of modifications and applications of this formalism into linguistics which are used in the filler-gap formalism in Chapters 3 and 4.

2.2 The Principles of Construction Grammar

Construction Grammar is a theory of grammar which assumes that meaningful units of language from the phonological to the semantic/pragmatic levels are composed of grammatical constructions, defined as an arbitrary and conventional pairings of form (signifier) and meaning (signified) (Goldberg 2013: 3-4). It diverges from most theories of generative grammar in that it handles individual words and syntactic and semantic elements in continuum rather than as distinct categories governed by different "rules" and posits a lexicon of constructions at all levels of linguistic complexity. Construction Grammar was first developed by George Lakoff, Charles Fillmore, and Paul Kay at the University of Berkeley in the 1980s in response to the popularization of Universal Grammar and its derivatives, which conversely proposed that the conventions of syntax in particular are innate in humans, and the process of acquisition requires learning which of a finite set of innately known syntactic parameters are present in the native language, rather than learning entire syntactic constructions as a relationship between form and meaning as one might learn vocabulary (Croft 2001: 263). However, while several linguistic definitions of theories in the family of Construction Grammars have been developed, more rigorous, formal mathematical definitions of meaningful grammatical "constructions" that could be used in parsers, natural language generation, and other computational applications have been relatively limited. In the following section, I will outline the basic attributes of a Construction Grammar and describe the advantages this theory may have over the canonical Chomsky grammars. These basic attributes are (Hoffman 2013: 2-4):

1. Constructions are form-function pairs which constitute the lexicon of a language. These lexemes include not only morphemes and words as in UG, but also syntactic structures and semantic features. CxG posits that all lexemes are acquired as "vocabulary."

2. Constructions do not form derivational relationships with one another; that is, one construction cannot be destructed to form another. Instead, constructions form inheritance relationships, wherein a subtype construction contains all the features of a supertype construction and can operate in place of the supertype.

3. There is no distinction between surface and deep structures. Construction Grammar has no move operations.

4. Just as each language has unique vocabulary (but may share words with other related languages), each language has unique constructions but may share constructions with related languages.

The first and foremost principle of Construction Grammar is that the lexicon of learned form-function pairs (i.e. "constructions") includes phrasal and clausal items as well as the conventional morphology and vocabulary. The reasoning behind this approach is that it can account for low-frequency forms as well as irregularities such as idioms which do not follow a so-called "grammatical" syntactic pattern. For example, while certain idioms such as "kicked the bucket" follow syntactic conventions but carry a semantic interpretation distinct from the lexical components, other idioms such as “all of a sudden” do not follow syntactic conventions and yet are parsed as grammatical. Furthermore, a standard UG approach proposes a restrictive set of phrase structure rules used to compose words into meaningful sentences. It then manipulates the surface-level syntax of sentences which do not follow these basic rules (such as wh-sentences and passive voice) into a “deep structure” via increasingly complex transformation operations, empty nodes, and traces. On the other hand, CxG instead proposes that these alternative forms are separately acquired structures, and they contain semantic information distinct to the structure regardless of the exact lexical items selected. To give an example, the standard approach proposes that “who did you meet?” is formed from a set of phrase structure rules which combine the four individual words into “did you meet who” and then the application of a move operation that creates “who did you meet.” The CxG approach instead proposes that a passive transitive-verb sentence is a distinct construction from a simple active transitive-verb sentence, that these constructions are acquired in childhood just like vocabulary words, and that they contain construction-specific meaning regardless of the individual lexical items substituted into the construction (Hoffman 2013: 2-6).

Of course, it cannot be true that every unique syntactic structure is individually acquired. There are infinite possible sentences which can be generated in most languages, and young children develop language competency with exposure to a limited set of such sentences, so a descriptive grammar must contain some features of recursion and substitution in order to produce grammatical sentences that one has not already heard. This is where the second principle of Construction Grammar applies. CxG maintains the idea that constructions may be composed of several other constructions, but it differs from mainstream generative grammar in that constructions form either inheritance relationships or compositional relationships with one another, rather than derivational relationships (Lichte 2017: 205-206). Every construction must either be defined as its own atomic type which cannot be further broken down into other types (generally a morpheme), as a linear composition of other types (for example, a PP is composed of a preposition followed by a DP), or as the subtype of another type, whose features the subtype must inherit (a subtype contains all the features of its supertype, but also contains additional constrains). For example, a DP with a modifier ("The big dog") is a subtype of the DP since the DP with modifier may be used in exactly the same ways and positions as the general DP, but with the added adjective/adverb parameter that the general DP does not specifically require. In contrast, a derivational relationship between constructions means that a construction may be destructed to form another—as we saw from the toy example of the passive voice, the standard approach maintains that passive syntax is derived from active syntax. Since constructions may not be derivationally related, CxG instead addresses the similarity in semantics between the active and passive due to the fact that both constructions are composed of similar smaller elements (i.e. the agent, patient, and transitive verb) (Goldberg 2013: 6-9).

These two features of CxG, namely signifier-signified pairings and inheritance/ compositional relationships, are the most important to consider when constructing the model described later in this paper. However, this chapter will also briefly address the remaining two principles which are included in most definitions of CxG. Firstly, CxGs should not require any transformations beyond the surface-level form to be interpreted semantically. Not only do the constructions themselves (without specific lexicalization) carry meaning, they also ensure that no rearrangement of lexemes into a "deep structure" is necessary to interpret the lexicalized construction. This principle mostly follows from the previous two. Constructions can only be generated via composition or inheritance from other constructions which means that there can be no longitudinal relationship between two forms with the same components in a different order. Since each individual construction is a pairing between form and meaning, ultimately, no equivalent "deep structure" is necessary to determine the meaning of a surface-level structure (Goldberg 2013:9).

Based on the given ideas the final principle also follows naturally—constructions are not expected to generalize cross-linguistically. In fact, CxG posits that each natural language has a unique set of constructions in its individual lexicon, where this lexicon of course includes constructions at the phrasal and clausal levels as well as individual words. To account for grammatical similarities between languages one might point out that related languages can have similar patterns of word order just as they have similar and/or overlapping vocabulary. The most basic constructions such as verbs are present in most natural languages since the concept of an action is a nearly universal meaning to which most languages can be expected to ascribe a form. CxG further allows us, for example, to consider languages which do not follow some of the most common syntactic and semantic patterns, such as a precise distinction between nominal and verbal forms, as simply using different constructions rather than as corner-cases or exceptions to an otherwise "universal" understanding of syntax (Croft 2001: 29-32).

Synthesizing the principles of Construction Grammar, George Lakoff provides a definition of a grammatical construction which will be used throughout the rest of this paper. A grammatical construction is defined either as an atomic, which is the smallest unit of meaning which cannot be decomposed further, or as a parent node representing the meaning of the construction itself and a set of child nodes representing the constructions which are composed to yield the parent. The parent node contains all semantic information relevant to how each of the component constructions are interpreted in relation to one another. Each of the child constructions itself consists of a parent node defining its own semantics, and a set of grandchild nodes which compose that child node. The semantics of the construction may be given using any semantic network representation. The example below from Sag (2011: 504) demonstrates the basic intransitive verb construction, using the frame semantics representation.

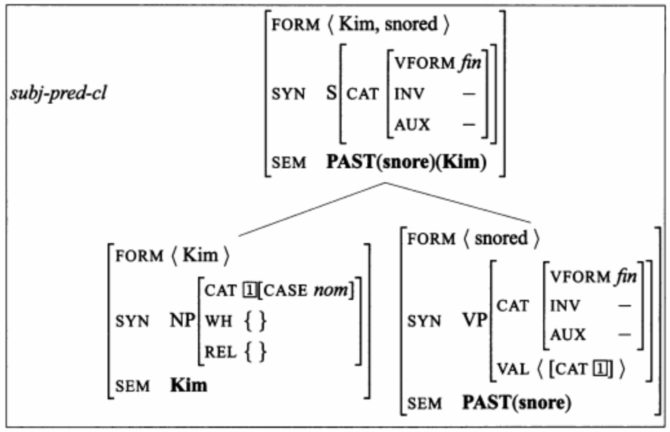


Figure 1: A construction representing an intransitive verb

2.3 Evidence For Construction Grammar: Past Literature

A classic example of the first principle of Construction Grammar comes from Adele Goldberg's characterization of the passive voice (Goldberg 2013). The canonical syntactic theory behind the passive voice is that the subject of a passive sentence originates in the complement of the main verb in the deep structure, and then, after the move operation, occupies the specifier of the tense phrase in the surface structure. Goldberg proposes an alternative CxG approach: the active and passive sentences are separate constructions, even if they contain the same combination of subject, object, and verb (for example "I ate the cookie" vs "The cookie was eaten by me"). This instantiation of the active sentence construction is composed of an agent (I), an active transitive verb (ate) and a patient (the cookie), whereas the passive construction is composed of the patient (the cookie), a passive transitive verb (was eaten), and a prepositional phrase containing the preposition "by" and the agent "me" (by me). While these two sentences have very similar meaning, the passive form prioritizes the patient's role in the action, while the active prioritizes the agent's role. This lends merit to the idea that the active and passive constructions represent a pairing between form and meaning that exists outside of the specific lexicalizations. Furthermore, Goldberg notes that several languages (including Choctaw, famously) do not have subject-demoting passives. Rather than treating these languages as outliers or exceptions, the CxG approach simply posits that the set of constructions that collectively defines all acceptable speech utterances in a language like Choctaw does not include a passive construction.

Goldberg (2013: 5-6) also points to experimental evidence for a direct form-function pairing between syntactic structures and semantic meaning regardless of the lexical elements which are placed into the "unfilled" slots. She notes that these structures largely retain their meaning when nonsense words are placed into unfilled slots. For example, in an instantiation of the ditransitive construction (simplified here to [Subj V Obj1 Obj2]) using the nonsense word "moop" as in "She mooped him something," native speakers interpreted the nonsense word as "give" 60 percent of the time and as some form of "literal or metaphorical" transfer (i.e. "tell") in all other cases [2]. This suggests that the syntactic structure [Subj V Obj1 Obj2] does not draw its meaning only from the lexical items substituted into the syntactic nodes but carries a semantic interpretation of transference itself which explains why this construction restricts the meaning of the smaller constructions placed into the verb slot to those with a sense of transference (i.e. "ditransitive verbs"). Goldberg further describes the phrases "She crutched him the ball" and "She crutched him." The first carries the interpretation that she used the crutch to transfer the ball to him, while the second carries the primary interpretation that she hit him with the crutch. Therefore, Goldberg claims that the interpretation of these two sentences, in practice, cannot come from the semantics of the individual words alone, since the word crutch is a noun which generally describes a physical object, not an action, yet the action of hitting is assumed [2]. Rather, the constructions themselves carry a semantic interpretation which further restricts the words which can be placed into the various subconstructions (i.e. I cannot "house" him a ball, since the word house describes a concept which cannot be used for striking).

Kay (2013) describes example of the second principle, the inheritance relationship: the P N construction which appears in phrases such as "They went to school," "She went to bed," or "They went on vacation." It is a subtype of the P NP prepositional phrase construction, in that every P N is a prepositional phrase (it meets all the requirements of a prepositional phrase) but of course there are P NP phrases which are not P N (consider “I went to my friend’s house”). However, we also note that not every noun can be placed into a P N (i.e. we cannot say “I went to store” or in American English “I went on holiday”). Therefore, we take the specific phrase “to bed” to be a phrasal lexeme combined from individual lexemes “to” and “bed” the same way morphemes “book” and “store” might combine to “bookstore” which has semantic interpretation coming from both the individual components and the unit. These phrasal lexemes must be acquired individually since there is no exhaustive rule to determine which words may form P N and which may not. Therefore, our inheritance relationship is: “to bed” is a subtype of P N which is in turn a subtype of P NP. We can then apply “to bed” in places which require a general P NP even if we have never heard “to bed” as the specific P NP used in that situation.

As mentioned in the previous chapter, Ivan Sag (2010: 507-511) presents a compelling case for Construction Grammar through analysis of filler-gaps, which is the case study selected for this paper as well. Sag identifies the two primary issues with the wh-movement approach: syntactic generalization over structures with distinct semantic values, and over-generation of ungrammatical speech. He further analyzes these shortcomings in terms of the principles of Construction Grammar. That is, the issue of generalization can be solved via application of the first principle of Construction Grammar. Each of the filler-gap constructions that he identifies are distinct syntax-level constructions with semantic value embedded in the constructions themselves. That way, a speech utterance which matches a wh-question will have an interrogative interpretation regardless of the specific lexical items, whereas a wh-exclamative will have an exclamatory interpretation, and so forth. Furthermore, the second principle of Construction Grammar can eliminate the issue of over-generation presented by the move operation. Since each of the individual constructions that he presents are defined via Lakoff’s model as the combination of a set of child constructions with a further set of semantic constraints, the CxG approach is able to specify exactly which wh-elements may be used for each construction, which syntactic categories the child constructions for the filler, main clause, and gap may belong to, the semantic value of the child constructions etc., such that no ungrammatical sentence may be produced. Since filler-gap constructions are so illustrative of the principles of Construction Grammar, they will be the primary case study presented in Chapter 4.

Very little research has been done to provide a complete formalization of Construction Grammar or to exhaustively define the set of grammatical constructions in any given language. The only notable work in this area is Fluid Construction Grammar, developed by Luc Steele. In its most recent iteration (2017), Fluid Construction Grammar (FCG) is a constraint-based formalism and a corresponding open-source software which linguists may use to experiment with constructions that they have defined. FCG takes in an input of either a speech utterance for parsing or a frame-based semantic network for speech production. Its repository of constructions provide a mechanism for mapping an input speech utterance to a semantic network, or the input semantic network to a speech utterance. Parsing and production become search problems, as the algorithm attempts to find the construction that best matches the input and returns the speech or semantic network produced by the construction that most closely matches the input. The constructions are modeled as operators which may be applied to input structures to produce output structures (Steels 2017: 178-182). The FCG formalism has been used by several linguists such as Remi van Trijp, who used it to compare two conflicting approaches to auxiliary verbs, namely treating them as VPs which can take NP complements, or as feature carriers (van Trijp 2017: 251-252).

For the purposes of comparing the UG and CxG approaches to filler-gaps, this paper employs the TAG formalism rather than the FCG formalism for two reasons: firstly, TAG has been identified as a potentially strong choice for adapting to Construction Grammar (see the following section) in recent years but has not been applied thus far, and secondly, since the stated goal of this research is to assess the CxG approach in comparison to the UG approach, a tree-based formalism similar to those already used by UG grammarians provides a straightforward way of comparing how sentences are composed in the opposing paradigms. Like the FCG formalism, the TAG formalism may, in the future, be used in an engineering context to experiment with newly-developed constructions in the context of parsing and speech production. The following subsection introduces the original TAG formalism and its adaptations which will subsequently be applied to Construction Grammar to evaluate filler-gap constructions in comparison to wh-movement.

2.4 Tree Adjoining Grammar

Tree-Adjoining Grammar (TAG) is a formalism first described by Joshi, Levy, and Takahashi in 1975 for its mathematical properties, but which has since been used as a meta-grammar for various theories of linguistics. This chapter first offers a brief description of TAGs and then describes how they have been used to characterize other linguistic theories and why they can be used as a tool to rigorously define the set of filler-gap constructions in English under CxG.

A simple TAG as defined by Joshi et. al. is a pair $G = (I, A)$ of two finite sets of elementary trees (Joshi 1985: 7-8). The first set I contains initial trees: all initial trees α have a head node containing some initial element or "start symbol." In the context of a X-Bar syntax tree, for example, an initial tree would be a tree rooted in "S." All leaves or "frontier nodes" of initial trees contain terminal elements, which do not need to dominate another element to exist in the grammar. The internal nodes are non-terminal. Again in the context of an X-Bar tree, terminal nodes would contain lexical elements, rather than syntactic categories such as "VP" or "Adj." Note that the following figures are taken from Joshi's paper.

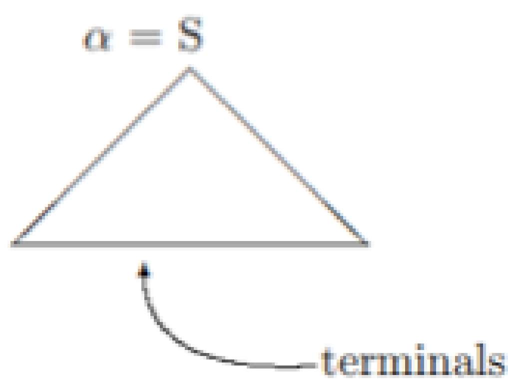


Figure 2: The structure of an initial tree

The second set A contains auxiliary trees: all auxiliary trees β are rooted in a head node containing a non-terminal element X. They must also have a frontier node containing that same non-terminal element X; this is called the foot node. All remaining frontier nodes of β must be terminal elements. As is true for initial trees, the internal nodes contain non-terminal elements.

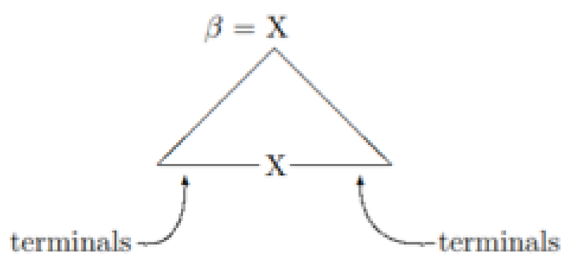


Figure 3: The structure of an auxiliary tree

There are no further restrictions upon the structure of initial and auxiliary trees, but in general these elementary trees will be minimal—that is, if tree E is in the set of elementary trees (either initial or auxiliary), it should not be possible to produce it by combining two other elementary trees via the adjoining operation described below, or else its inclusion in the set of elementary trees is redundant.

Elementary trees are combined with one another recursively using an operation called adjoining (hence Tree-Adjoining). Through adjoining, a node containing a non-terminal element is replaced recursively by an auxiliary tree whose head and foot contain that non-terminal element as follows (Joshi 1985, 8-10):

1. Tree G_1 contains some non-terminal element T, with parent a_0 and children $a_1 \ldots\ a_n$. The sub-tree A rooted at T is removed from G.

2. Tree B is an auxiliary tree with head and foot containing element T. Tree B is inserted into tree G_1 such that a_0 is the parent of B—that is, Tree B has replaced the sub-tree A in G_1.

3. The foot node containing element T is removed from B and replaced with sub-tree A. That is, since sub-tree A's head is T, it replaces the foot node of B which also is element T.

Consider these elementary trees: initial tree γ_1 and auxiliary trees β_3, and β_4 (Figure 4).

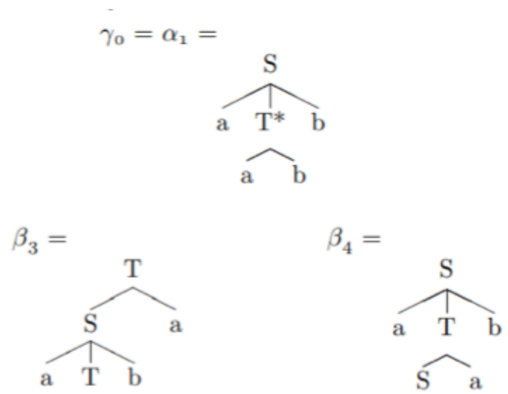


Figure 4: The elementary trees

If we are to adjoin β_3 to γ_1 at the node containing a non-terminal element T(the specific element marked as follows as T^*), first the sub-tree rooted at T^* is removed from γ_1 (Figure 5).

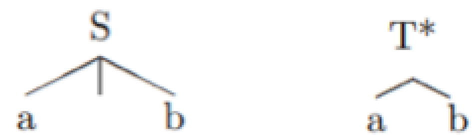


Figure 5: Removing the subtree rooted at T*

Next, tree β_3 is inserted into γ_1 in the position from which the sub-tree was removed (Figure 6).

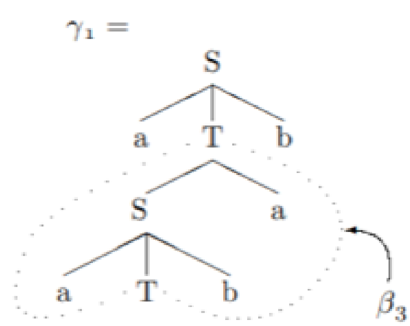


Figure 6: Inserting the auxiliary tree

Finally, the sub-tree rooted at T^* is inserted back into γ_1 at the foot of β_3, replacing the T element already there (Figure 7).

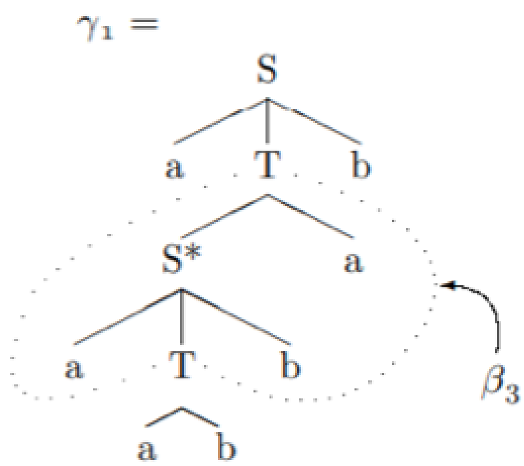


Figure 7: Re-inserting the removed element.

Next we will adjoin tree β_4 into γ_1 at a non-terminal node containing S (identified as follows as S*) (Figure 8).

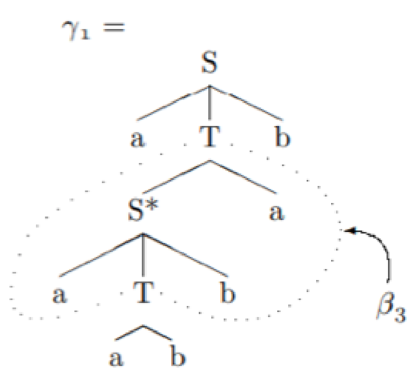


Figure 8: The beginning tree

First, we will remove the sub-tree rooted at $S*$ from γ_1 (Figure 9).

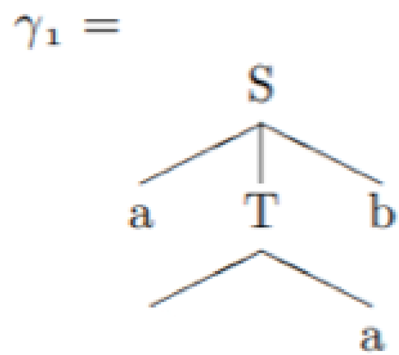


Figure 9: Removing the sub-tree rooted at S*.

Next, tree β_4 is inserted into γ_1 at the position vacated by the sub-tree rooted at $S*$ (Figure 10).

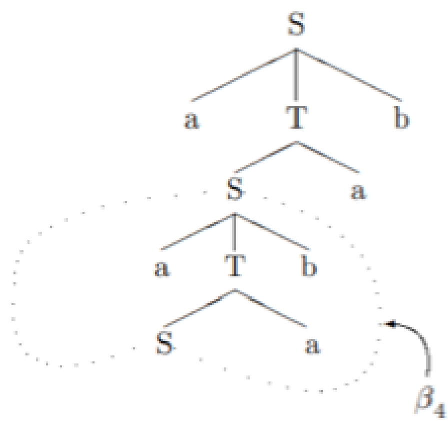


Figure 10: Inserting the auxiliary tree.

And finally, the sub-tree rooted at S^* which was removed from γ_1 is placed back into the tree at the foot of β_4.

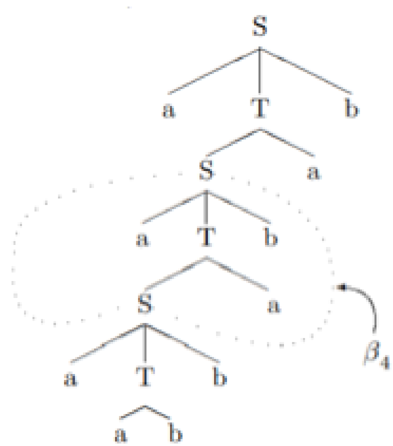


Figure 11: Replacing the sub-tree rooted at S*.

The first adjoining operation that was performed combined two elementary trees: an initial tree with an auxiliary tree. The second operation, however, shows that the adjoining operation can be performed upon non-elementary trees as well. After adjoining γ_1 and β_3, the resulting tree was a non-elementary tree. We were then able to adjoin β_4 into that non-elementary tree. However, the element adjoined into any elementary or non-elementary tree is always an elementary auxiliary tree.

2.5 Variations On Tree-Adjoining Grammar

2.5.1 Linked Tree-Adjoining Grammar

Prior research has proposed several variants of the basic Tree-Adjoining Grammar with the specific purpose of linguistic applications. The first of these modifications is Linked TAG, developed by Joshi et al in 1985, which is of particular relevance to filler-gap constructions. Linked TAG introduces a new relationship between nodes within a single tree called a "link."

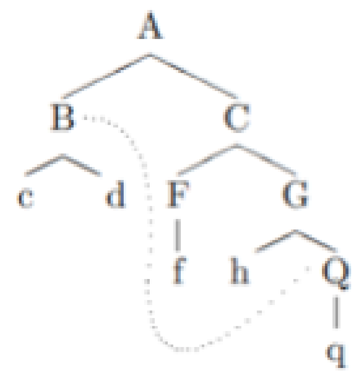


Figure 12: A linked TAG tree.

In Figure 12, the nodes containing elements B and Q are linked, as demonstrated by the dotted line. The link relationship between nodes n_1 and n_2 can exist under two conditions:

1. n_1 must c-command n_2—that is a sibling of n_1 must dominate n_2

2. n_2 only dominates nodes containing null or terminal elements.

To illustrate these two conditions, note that node G may not be linked to node Q (G does not c-command Q) and B may not be linked to G (G dominates non-terminal node Q). Furthermore, the TAG link is a directed link (meaning that if n_1 is linked to n_2, n_2 is not necessarily linked to n_1). Note that while B is linked to Q, Q cannot be linked to B since Q does not c-command B.

The link relationship is preserved under the adjoining operation. That is, the links remain between the precise elements n_1 and n_2 after the adjoining and do not get transferred between nodes with the same non-terminal element. That is, in the above example, if an auxiliary tree with head and foot containing Q were adjoined into the given tree, the link would remain between B and the Q which is the parent of q, not the newly inserted Q which is the child of G and sibling of h. We illustrate with the following example:

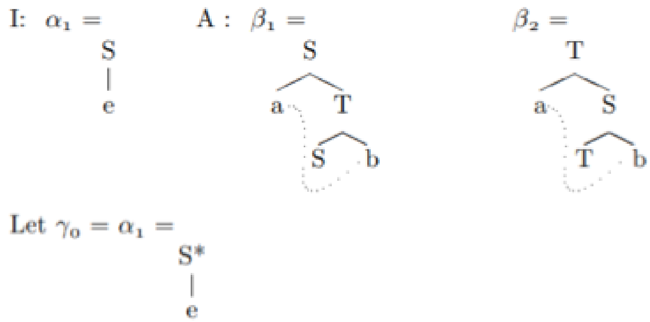


Figure 13: A set of elementary linked TAG trees.

Then if we are to adjoin tree β_1 with α_1, we end up with:

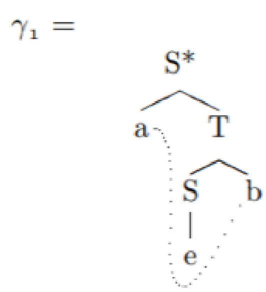


Figure 14: Linked tree after adjoining.

The link between a and b is preserved when we adjoin S^* to the S in the non-terminal node in β_1.

2.5.2 Lexicalized TAG (LTAG)

A 1998 project from the XTAG Research Group at the University of Pennsylvania developed another variant on Tree-Adjoining Grammar called Lexicalized Tree-Adjoining Grammar (LTAG). A lexicalized grammar is a grammar which defines a finite set of structures "anchored" by a lexical item—historically this lexical item has been a word. It must also propose a finite set of rules by which the lexical items may be combined. Therefore, LTAG is a modification of the TAG formalism in which each elementary tree is anchored by a lexical item. The finite set of rules by which the lexical items are combined includes the basic adjoining operation as well as a new operation called substitution. The formal definition of an LTAG is a 5-tuple $G = (\Sigma, V, S, I, A)$ where:

1. Σ is the finite set of all terminal symbols (for example words), analogous to the terminal symbols in basic TAG

2. V is the finite set of all non-terminal symbols or variables, analogous to the non-terminal symbols in basic TAG

3. S is the designated start symbol as in basic TAG. Not all initial trees necessarily start with S as the root.

4. I is the finite set of all initial trees as in basic TAG with some differences. An initial tree in LTAG may begin with any non-terminal symbol (not necessarily the start symbol S). Of its frontier nodes, one must be a terminal symbol (the anchor). The remaining frontier nodes may include other terminal symbols or non-terminal symbols. All non-terminal frontier nodes must be marked for substitution. All internal nodes are non-terminal, as is the case for basic TAG.

5. A is the finite set of all auxiliary trees. Like with the initial trees, auxiliary trees must be anchored by some terminal frontier node. The remaining frontier nodes may either be non-terminal nodes marked for substitution, or terminal nodes. The internal nodes are all non-terminal.

This definition is adapted from Penn Linguistics' website (2020). Note that for both I and A, as with basic TAG, minimality of elementary trees is not required by definition but is generally implemented to avoid redundancy.

The adjoining operation to recursively insert any auxiliary tree into any other elementary or non-elementary tree remains the same between TAG and LTAG. However, LTAG introduces the substitution operation, which serves to add children to a non-terminal frontier node. That is, in the example of a syntax tree, if a frontier node contains an empty NP (i.e. simply the syntactic category with no actual lexical nouns, determiners, etc included, perhaps the tree “DP barked”), a tree containing a full DP (for example “The dog”) may be substituted into the first tree.

More formally, the substitution operation is an alternative way of combining lexicalized trees which does not require the insertion of an auxiliary tree. Given any tree rooted in element A and any other tree with a frontier node A marked for substitution, the frontier node A may be replaced by the tree rooted in A, as seen in Figure 15 (from Joshi 1985).

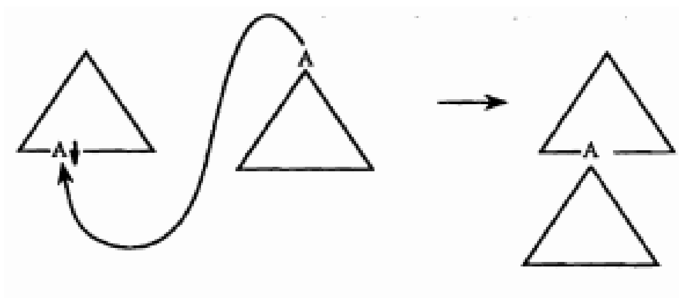


Figure 15: The substitution operation

In the context of a syntax tree with lexical items, the substitution operation is helpful for replacing syntactic elements (VP, NP, etc) with lexical items that fit into that syntactic element. That is, if the initial tree α was the syntax tree for the intransitive verb “sleeps,” the non-terminal node DP would be marked for substitution. Then, the elementary tree σ containing the proper noun “Jonathan” which is rooted in a DP node could be substituted into α to produce the tree for “Jonathan sleeps.” That is, the substitution operation helps us to insert lexical items into a tree formed with syntactic categories, in other words, non-recursive language generation.

On the other hand, recursive generation is produced by the adjoining operation. The adjoining operation remains the same as with basic TAG, with the clarification that an auxiliary tree may only be adjoined at a non-terminal node NOT marked for substitution. Furthermore, the auxiliary tree being adjoined may also have several non-terminal nodes, but all except the foot (which matches the head) must be marked for substitution and therefore cannot be used for adjoining. We can demonstrate both of these properties by describing the basic phrase structure tree of a simple sentence "John always walked." If the NP for "John," the VP for "walked," and the VP for "always" as shown in Figure 16 are all valid trees of a TAG, then we can combine "John" and "walked" into "John walked" by substituting in the NP "John" for the empty NP in the VP "walked." We can combine "John walked" and "always" by adjoining the VP of "always" to the VP node in "John walked" which contains the verb itself. The VP "always" has an empty VP as a foot, and we replace the intermediate VP which we removed from "John walked" at that foot, to end up with "John always walked." This example is adapted from Lichte (2017).

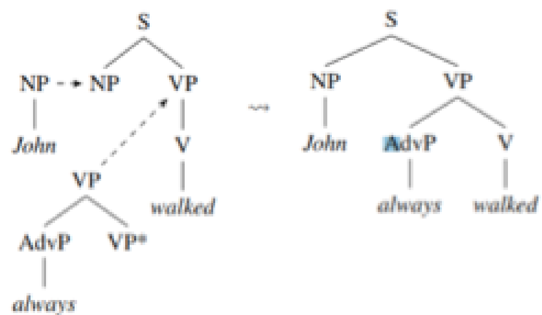


Figure 16: Lexicalized Adjoining

2.5.3 Constrained TAG

Nesson and Shieber (2006) also demonstrated that we can limit the types of operations which can be performed on a given node by introducing local constraints. Selective adjoining occurs when only some specific subset of auxiliary trees can be adjoined at a given node. For example, if a non-terminal node n is labeled X but with constraints that only trees in set S may be adjoined, even trees with head and foot X which are not in set S may not be adjoined at n. Obligatory adjoining states that for some node n, there exists some valid auxiliary tree which must be adjoined at n. Finally, null adjoining is a special case of selective adjoining where the subset of valid adjunctions is empty—no auxiliary tree may be adjoined at that node regardless of its head/foot.

Constraints must also be unified after each substitution or adjunction. This means that if node X with constraints C_1 is replaced in a tree by node Y (and its descendants) with constraints C_2, then node Y is now constrained by the union of C_1 and C_2. The operation cannot be completed if the token in that node is unable to satisfy both sets of constraints.

2.5.4 TAG's Potential In Construction Grammar

Lichte and Kallmeyer (2017) have speculated about potential ways in which TAGs could be used to formalize the central tenants of CxG which Goldberg has described. First and foremost, TAGs are naturally suited to the notion of "surface-level" grammar—no constructions may be decomposed to create other constructions, and all non-atomic constructions are inherited from or composed of other constructions. The simplest definition of a TAG has no obvious way of performing move/trace operations as they are defined in most generative grammar theories. These move operations require an element to be placed into a tree at a certain point and then subsequently detached and moved to another part of the tree, leaving behind a trace. The example in Figure 17 illustrates this point using the wh- question "Who does Mary say walked into the house?" A grammar which includes movement would instantiate the auxiliary tree "does… say" on the same level as "walked" and then perform a move operation to leave a trace in that position and move that tree up to the head. Modeling this sentence with TAGs instead produces the correct sentence by adjoining the independently generated auxiliary tree directly into its surface-level position in the sentence. The right hand tree represents a construction that takes parameters of certain wh- words (in this case who or what), a phrase with an auxiliary verb, and a verb phrase (each of which is its own construction), and the full sentence is constructed by adjoining the parameter constructions into the largest construction, with no consideration of any theoretical deep structure from which this sentence might be derived. Of course, this is a simplification of the specific details of what may or may not be included in each of the positions of the largest construction, but these constraints can certainly be defined as described above via selective and obligatory adjoining (Lichte 2017: 207-208).

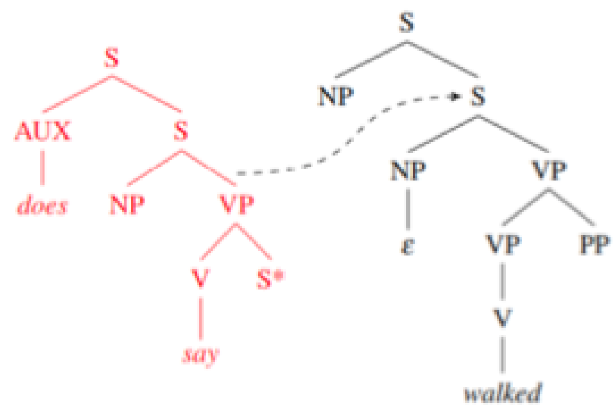


Figure 17: Potential TAG approach to Construction Grammar

Now that the basics of Construction Grammar, its potential applications to filler-gap sentences, and the basics of Tree-Adjoining Grammar have been introduced, Chapters 3 and 4 will explore the application of TAG to the principles of Construction Grammar to develop a set of filler-gap constructions.

3 Construction Grammar With A TAG Formalism

3.1 Introduction

This chapter will provide a description of how to adapt the Tree-Adjoining Grammar formalism to represent the principles of a Construction Grammar. It will provide definitions for a "grammatical construction" and a "Construction Grammar" for a given language as sets of elementary trees, and it will define the inheritance and compositional relationships between constructions in terms of these sets. It will also address the finite and recursive elements of a Construction Grammar using the substitution and adjunction operations of the TAG formalism. This implementation of the principles of Construction Grammar in the TAG formalism will form the basis of the primary research in this paper presented in Chapters 4 and 5, which define and analyze filler-gap constructions in comparison to the wh-movement paradigm.

3.2 Elementary Trees

To begin with a note on convention, the terminal nodes of TAG trees will be indexed in the order of verbal speech. That is, in the sentence "I went home," the terminal node anchored by "I" will be n_0, the terminal node anchored by "went" will be n_1, and the terminal node anchored by "home" will be n_2.

Definition 3.1. *A Speech Utterance l can be represented as an ordered set $\{n_0 \ldots n_n\}$ of words, where the words are indexed in the order that they are spoken.*

The above definition simply serves to represent speech in set notation for the purposes of defining grammaticality in a Construction Grammar, as follows:

Definition 3.2. *A Construction Grammar for a language L is defined as a lexicalized, linked, and constraint-based Tree-Adjoining Grammar $G = (I^*, A^*)$. For all valid utterances l in L, there exists a tree T produced via adjunction or substitution of elementary trees in G, whose terminal* *nodes form an ordered set* $\{n_0 \ldots n_n\} = l$*. For all utterances* l_u *not in* L*, there are no trees* T *which can be produced via adjunction or substitution of elementary trees in* G*, whose terminal nodes form the ordered set* $\{n_0 \ldots n_n\} = l_u$*.*

Essentially, grammaticality has been defined in terms of which utterances can and cannot be produced via TAG operations from the set of elementary trees. If a speech utterance is grammatical, it is either an elementary tree or some combination of elementary trees acceptable under TAG rules. Furthermore, application of TAG rules to elementary trees cannot produce an ungrammatical speech utterance. Note also that while TAGs are defined with respect to a finite set of elementary trees, the adjunction and substitution operations allow for the generation of infinitely many non-elementary trees via recursive application of elementary trees (see section 3.3).

Definition 3.3. *A 'Construction' C belongs to the grammar G if C is a nonempty set, and C is a subset of either I** *or A***.*

Constructions are sets of elementary trees, which generally have some shared properties with one another. Naturally, then, all constructions must be subsets of either I^* or A^*. For example, the lexical constructions "dog" and "cat" both belong to the N (noun) construction. Constructions are not sets of rules for combining simple trees into more complex trees. Rather, they simply identify which elementary trees have similar semantic and syntactic properties. Note that many different constructions may be anchored by the same lexical element. For example, the word "eat" may anchor both the transitive verb construction and the intransitive verb construction. Therefore, when parsing with the TAG approach, each word may instantiate several different constructions, and the parser must select the correct construction for each word, such that the constructions are able to combine following TAG rules into the full speech utterance.

As a matter of convention, initial trees will be rooted in the name of the construction to which they belong. Returning to the example above, the elementary tree for "eat" as a transitive verb phrase is rooted in VP-trans and anchored in "eat" while the elementary tree for "eat" as an intransitive verb phrase is rooted in VP-intrans but anchored in "eat" as well. Auxiliary trees, however, must be rooted in the type they modify so that they may be adjoined above those types.

Finally, constructions may contain either initial trees or auxiliary trees, but not both. As is explained further in section 3.3, if a terminal node of any TAG tree is marked for substitution by a construction C, then any elementary tree which is a member of C may be substituted into that position. Recall that the specifications of LTAG state that auxiliary trees cannot be adjoined at nodes marked for substitution. Therefore, if a construction contains both initial and auxiliary elementary trees, not all members of the construction could be substituted into the same position, which contradicts a basic property of constructions.

Definition 3.4. *A construction C_2 is said to inherit from another construction C_1 if $C_2 \subseteq C_1$.*

The inheritance relationship from a linguistic standpoint means that the subtype construction may be used in any situation that calls for the supertype construction, but that certain situations may call for the subtype construction specifically, but not necessarily all other instantiations of the supertype construction. Therefore, this definition simply means that every elementary tree in the subtype construction C_2 must also be in the supertype construction C_1. If any node in a TAG tree is marked for substitution with the supertype C_1, any elementary tree in C_1 may be substituted. Therefore, it follows that every tree in C_2 may also be substituted, since $C_2 \subseteq C_1$. Conversely, if a tree has a node marked for substitution with the subtype C_2, there may be trees in the larger set C_1 which are not in C_2 and cannot be substituted.

Definition 3.5. *Given construction C and the ordered set of constructions $C_s = \{C_0 \ldots C_n\}$, C_s composes C if, for all elementary trees c in C, for all immediate children (in order) $d_0 \ldots d_n$ of the root of c, d_i is an element of C_i.*

To further explain the definition given above, for any given set representing a construction, each member of that set is an instance of that construction. That instance must be a combination of some smaller elements, which are the immediate children of the root of that construction. What this definition suggests is that every instance of the construction must be composed of the same set of smaller constructions.

We say, linguistically, that a parent construction is composed of an ordered set of child constructions if combining members of that child construction in order produces a member of the parent construction. For example, we might say that a prepositional phrase is composed of a preposition and a DP, and combining any utterance that belongs to the preposition construction (for example, “in”) with any utterance that belongs to the DP construction (for example, “the kitchen”) produces an utterance that belongs to the prepositional phrase construction (“in the kitchen”).

3.3 Recursion

This chapter has thus far dealt primarily with sets of elementary trees and how those sets relate to one another in the context of grammatical constructions’ inheritance and composition relationships. However, language is not a finite set of utterances. An infinite number of sentences are theoretically possible—consider, for example, a sequence of prepositional phrases “I bought the apples in the fruit aisle, in the grocery store, in the town...” and so forth. The TAG formalism allows for infinitely recursive sentences via the two operations for combining elementary trees, adjunction and substitution. The following example shows how two elementary trees, one initial and one auxiliary, may be recursively combined into non-elementary trees. This instance deals with modifying a verb with an additional helping verb. Other structures such as adverbs or prepositional phrases migt also be adjoined.

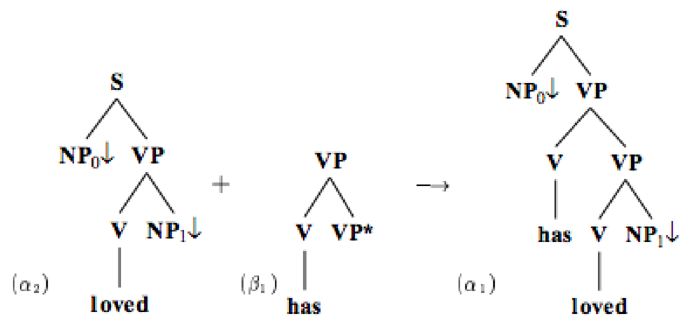


Figure 18: Recursion via adjoining

As a general rule, the types of constructions such as AdvP and AdjP which linguists call "adjuncts" are inserted into elementary trees via (unsurprisingly) the adjoining operation. The structures which linguists call "complements" are inserted into elementary trees via the substitution operation. To address the issue of inheritance and composition between non-elementary trees:

Definition 3.6. *If C_1 inherits from C_2, then for all c^* produced from TAG operations on some c in C1, c^* also inherits from C2.*

This amended definition ensures that adding a modifier or a complement to an elementary tree will not limit the places in which it can be substituted. For example, if "The dog" may be substituted into "DP barked," then "The big dog" may also be substituted into "DP barked." Going by the principles of constraint-based construction grammar, this infinite recursion can be limited in certain instances by syntactic or semantic category constraints placed on individual nodes in a tree. For example:

(3.1) Laura ate the cake.

(3.2) *Laura ate the imagination.

Although "the cake" and "the imagination" are both in the set DP, the transitive verb construction anchored by "ate" imposes semantic constraints upon the content of its complement. That is, of course, the verb "eat" can only accept objects which are edible. Therefore, while it seems that "the imagination" can be substituted into the complement of "ate" according to the basic rules of an LTAG, in truth the additional constraints prevent such a substitution from occurring.

3.4 A Note On Semantics

Since this is primarily a study of syntax, a complete description of how to represent a construction's semantic value in TAG will be elided. However, potential mechanisms of integrating frame semantics into the TAG representation of Construction Grammar will be addressed briefly in Chapter 5. For the purposes of this study, the semantic interpretation of a given construction will be described informally only. For example, the sentence "Whose cat is that?" is said to have an interrogative semantic interpretation.

With a mechanism of describing grammatical constructions in terms of the Tree-Adjoining Grammar formalism in place, Chapter 4 will now apply the contents of this chapter to a specific case study in filler-gap sentences.

4 A Case Study in Filler-Gap Constructions

4.1 Introduction

The goal of this section is to define a grammar for several types of filler-gap sentences using the TAG implementation of Construction Grammar presented in Chapter 3. As described in Chapter 2, grammatical constructions are often best described in terms of their inheritance and compositional relationships with other constructions. Each of the four categories of filler-gap constructions addressed in this paper– topicalized clauses, wh-exclamatives, wh-interrogatives, and wh-relatives, inherit from one of two supertype constructions, which will be named the Initial and Auxiliary Filler-Gap Constructions (FG-cxn-init and FG-cxn-aux). Every subtype construction which inherits from FG-cxn-init or FG-cxn-aux will demonstrate all the features of FG-cxn-init or FG-cxn-aux respectively, but will have additional unique specifications for the individual subtype, determining exactly which child constructions are composed in this specific subtype to meet the specifications of the supertype FG-cxn. This approach demonstrates the unique power of Construction Grammar to articulate generalizations across several different structures while also preserving the differences between each type. This chapter will first define the supertypes FG-cxn-init and FG-cxn-aux and visualize them using TAG. Then, a subtype construction will be defined for each of the four selected varieties of filler-gap sentence and implemented in TAG. Several examples will be used to demonstrate sentence generation using each of the defined constructions. Detailed justification for the approach taken in this chapter and assessment of its ability to prevent overgeneration will be addressed in Chapter 5 (Discussion).

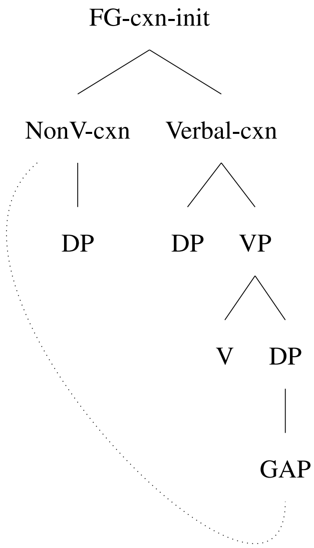
4.2 The Filler-Gap Constructions

A filler-gap sentence is characterized by two primary features: the “filler” clause, and the main clause, which contains a “gap.” The gap may be a complement, such as an unfilled verb valency, or an adjunct, such as an adverb phrase or a prepositional phrase, to a main verb in the main clause. Furthermore, the filler must always be of the same syntactic type and semantic category as the gap in the main clause, such that the filler is interpreted semantically as the complement or adjunct of the verb in the main clause. It is also notable that the filler and gap are always nonverbal. Therefore, all of the Filler-Gap Constructions must be composed of two daughter constructions, one for the filler and one for the main clause, where the gap in the main clause must be of the same nonverbal syntactic and semantic type as the filler.

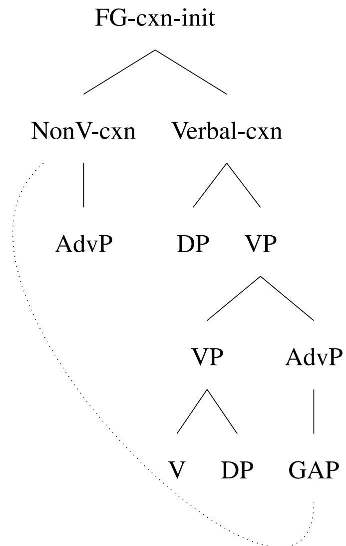
A linked TAG approach to representing the supertype FG-cxns-init will be headed by FG-cxn-init as is true of all initial tree constructions. It will have two daughters: the filler of type NonV-cxn, and the main clause of type Verbal-cxn. NonV-cxn is a supertype that accepts all nonverbal phrases such as AdvP, AdjP, NP, DP, PP, gerunds, and so forth. Verbal-cxn is a supertype that accepts all structures with a main verb, such as linear sentences (S-lin-cxn), inverted sentences(S-inv-cxn), and infinitives (inf-cxn). This ensures that every type of main verb may appear in some filler-gap construction, and that any type of adjunct or complement to the main verb may occur in the filler. Furthermore, the Verbal-cxn must also contain one complement or adjunct frontier node with a link drawn from the filler, specifying that these two elements must be of the same syntactic and semantic category. When substituting lexical elements into these trees, only the empty element "GAP" may be inserted into this linked complement or adjunct. More precisely:

Definition 4.1. *The Initial Filler-Gap Construction is defined as the set of all elementary initial trees composed of children of type NonV-cxn and Verbal-cxn. The Verbal-cxn contains an empty "GAP" element of the same syntactic and semantic type as the NonV-cxn and a link to the "GAP" element from the NonV-cxn. The GAP element can only occur as the complement or adjunct to an S construction.*

(4.1)



(4.2)



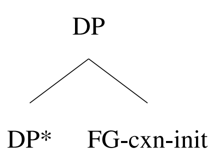
-

The set of FG-cxn-init elementary initial trees contains every initial tree that has some combination of NonV-cxn, Verbal-cxn, and adjunct/complement. Two examples are given above. In Figure 1, the filler is type DP, the main clause is a linear sentence, and the gap is the DP complement to the main verb. Such a tree might represent the sentence "Cookies, I will eat"(i.e. "But donuts, I won't."). In Figure 2, the filler is type AdvP, the main clause is a linear sentence, and the gap is the AdvP adjunct to the verb. Such a tree might represent the sentence "How quickly she read the book!" Any construction meeting these specifications can be said to inherit from FG-cxn-init, and therefore may be used in any position where the supertype FG-cxn-init is accepted for substitution. The most obvious example is the construction S (a sentence), which is a supertype of FG-cxn-init. Any tree built upon an initial tree in the set FG-cxn-init may be used any time an S construction is required or allowed, whether that is a standalone sentence ("What a good book that was!") or following a verb of saying, knowing, thinking, etc. ("I knew what a good book that was."), unless further constraints apply.

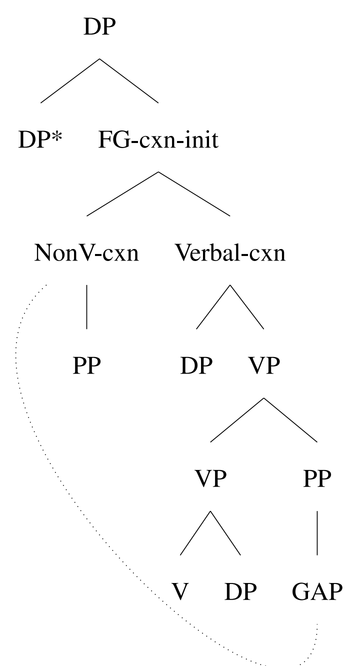
The TAG approach to representing the Auxiliary Filler-Gap Construction FG-cxn-aux follows similar logic to the Initial Filler-Gap Construction. These constructions do not inherit from S, and instead modify nominal constructions. For example, in the statement ”The person who you met is tall,” ”who you met” is an adjunct to the subject of the sentence, ”the person.” Therefore, this construction is not rooted in FG-cxn-aux, but rather in the DP which it modifies. All other features of the FG-cxn remain the same, in that it must have two additional children namely NonV-cxn and Verbal-cxn, Verbal-cxn must be anchored in the empty element ”GAP” which is either an adjunct or a complement to the main verb, and there must be a link between the filler NonV-cxn and the gap, which are of the same syntactic and semantic type. More formally:

Definition 4.2. *The Auxiliary Filler-Gap Construction is defined as the set of all elementary auxiliary trees with head and foot in DP, composed of children of type DP and FG-cxn-init.*

(4.3)



(4.4)



-

Figure 3 represents the general form of all elementary auxiliary trees which belong to the set FG-cxn-aux. These auxiliary trees modify nominals of type DP, and subsequently the first child must be type DP. The second child must be constructed from an elementary initial tree belonging to the set defined by FG-cxn-init. Figure 4 shows an example of an elementary auxiliary tree which belongs to the set FG-cxn-aux. This tree could represent an adjunct such as ”in which I bought the cake,” which might be adjoined to a sentence such as ”The store was closed” at the DP to produce ”The store in which I bought the cake was closed.”

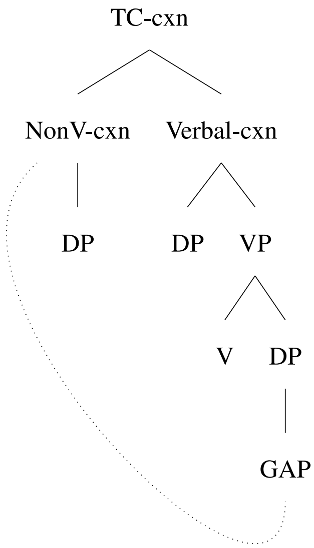
The following subsections will outline how restricting the types of NonV-cxn and Verbal-cxn which may serve as the filler and main clause in a FG-cxn results in unique sub-types that demonstrate four different semantic usages.

4.3 Topicalized Clauses

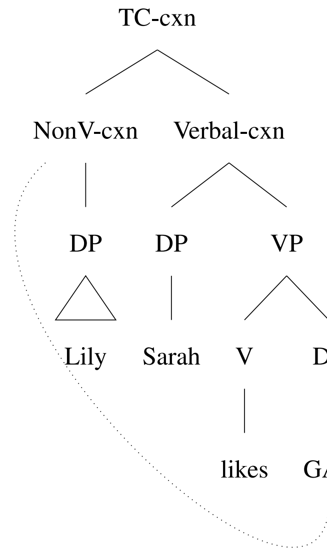
Topicalized clauses (TC-cxn) are the most basic filler-gap constructions. The filler may belong to any nonverbal category, but the main clause is limited to linear sentences (“I will go”), as opposed to infinitives (“to go”) or inverted sentences (“Will I go”). In other words, we may say “To the store, I will go” but we may not say “To the store, to go” or “To the store, will I go.” The gap may still be any adjunct or complement. The topicalized clause inherits from the sentence construction S as well as FG-cxn-init, meaning that it can stand on its own as an independent sentence. Finally, the filler may not belong to a wh-construction. Wh-constructions will be defined explicitly in the next subsection. Therefore, we may define the TC-cxn as follows:

Definition 4.3. *The Topicalized Clause may be defined as the set of all elementary initial trees in FG-cxn-init whose filler is not type wh-cxn, and whose main clause is type S-lin-cxn.*

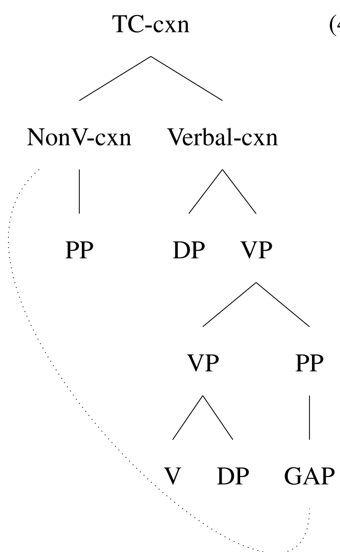
(4.5)



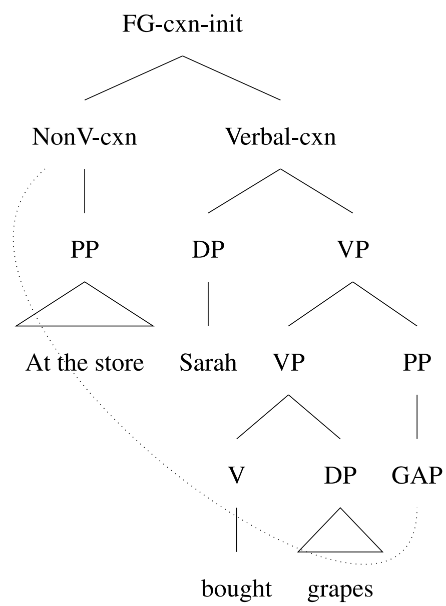
(4.7)



(4.6)



(4.8)

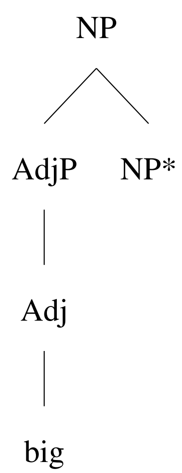


-

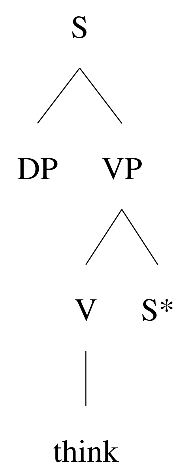
Figure 5 is an elementary tree in the set TC-cxn which contains a DP as the NonV-cxn, linked to the complement of the main verb. Figure 6 is an elementary tree in the set TC-cxn which contains a PP as the NonV-cxn, linked to the adjunct of the main verb. Figure 7 is a full lexicalization of Figure 5, and Figure 8 is a full lexicalization of Figure 6. These examples demonstrate that the topicalized clause may accept different types of NonV-cxn in the filler.

Topicalized clauses can be recursively modified by adjoining any auxiliary tree whose foot matches a node at any position in the TC-cxn initial tree. These include verbs of knowing, thinking, etc. as well as adjectives, adverbs, and prepositional phrases. By this logic, the sentence “At the store, Sarah bought grapes” can be modified to “At the big store, I think, Sarah quickly bought grapes” by adjoining the auxiliary trees for lexical elements “think,” “quickly,” and “big.” These elements are all modifiers, so they are entered into the grammar as auxiliary trees whose feet are the elements they can modify. “Think” modifies a sentence containing what the subject is thinking, “big” modifies a noun, and “quickly” modifies a verb. The following are the auxiliary trees representing these elements:

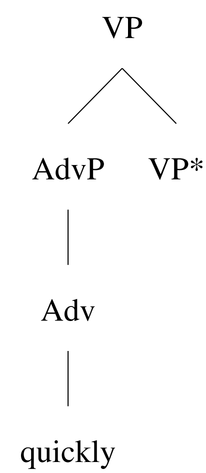
(4.9)



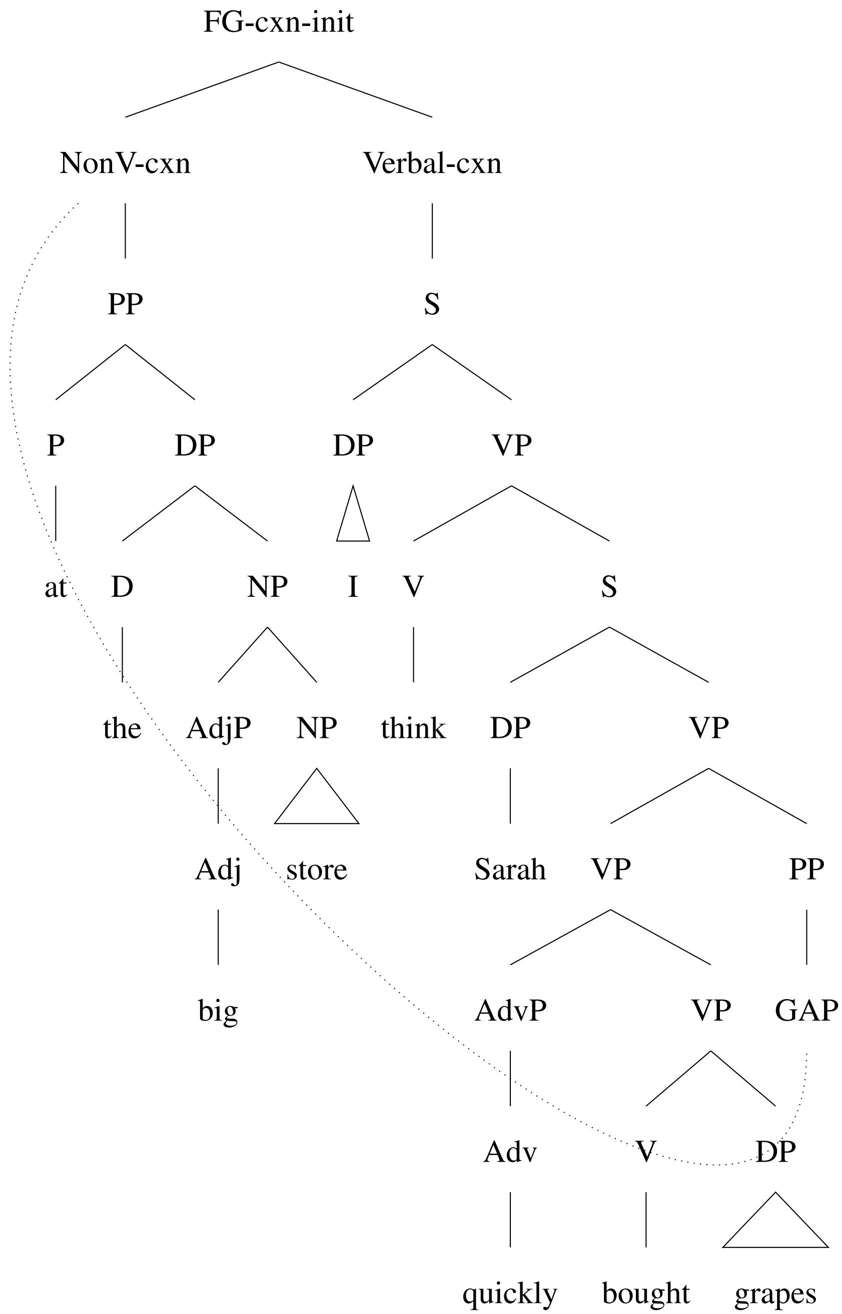
(4.10)



(4.11)



After adjoining to the tree in Figure 8 at the highlighted nodes, the resultant tree is:

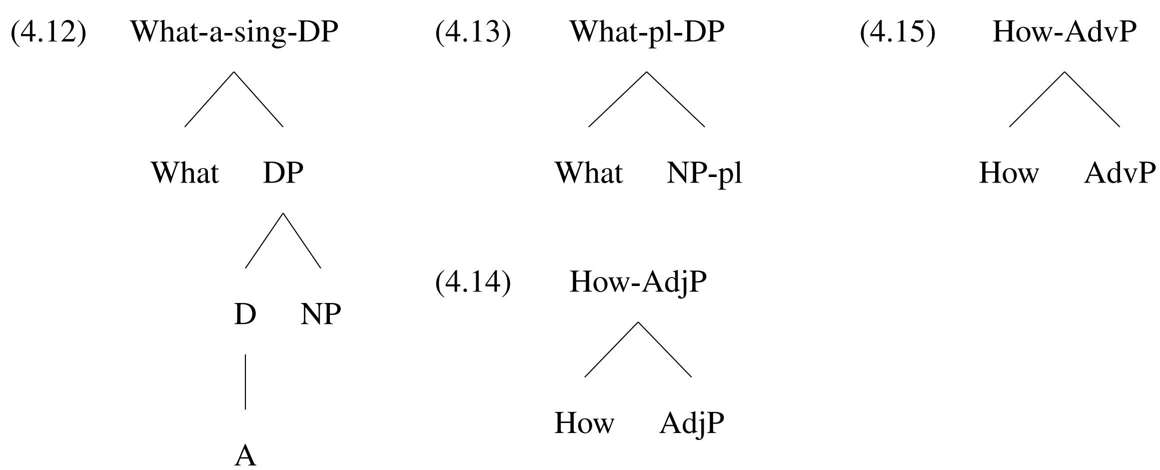


From the above examples, the semantic value of the TC-cxn is what Sag describes as an "austinean" value. That is, there is an implication of contrast between the topicalized filler and any other values which might have occupied the filler. Example 7 differs from the untopicalized sentence

"Sarah likes Lily," because the untopicalized sentence does not imply whether or not Sarah likes or dislikes anyone else. On the other hand "Lily, Sarah likes" carries the meaning of Sarah liking Lily, as opposed to someone else (i.e."Lily, Sarah likes, but Ben, she doesn't."). Likewise, "At the store, Sarah bought grapes" emphasizes that it was the store where Sarah bought grapes, as opposed to some other location (i.e. "At the store, Sarah bought grapes, and on the internet, she bought clothes.").

4.4 Wh-Exclamatives

Wh-exclamatives are the first type of filler-gap construction that uses wh-word constructions. Note that wh-words are not simply question words that begin with "wh," but can also include words such as "how," used in the wh-exclamative, and "that," used later on in the wh-relative. The wh-exclamative construction (Wh-ex-cxn) may accept What-pl-DP, What-a-sing-DP, How-AdjP, and How-AdvP, which are described as follows in Figures 1, 2, 3, and 4 respectively, as the nonverbal construction.

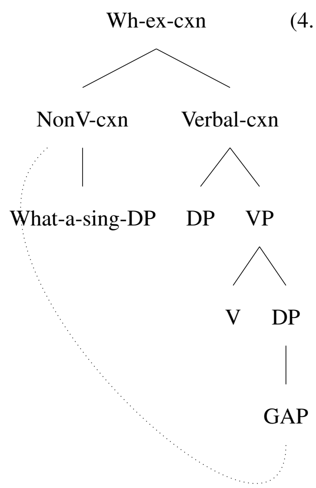


The "what" constructions both have the syntactic type DP and are subtypes of NonV-cxn. Likewise, How-AdjP and How-AdvP have the syntactic types AdjP and AdvP respectively, and also inherit from NonV-cxn. In construction grammar, all lexical elements are constructions, and these constructions define how the words may be combined with other words to form sentences. However, since some words have multiple meanings and may be combined with other words in different ways, each of these different forms is entered into the lexicon as a different construction. This is why What-a-sing-DP and What-pl-DP are distinct constructions for different uses of the word "what," and why How-AdjP and How-AdvP are distinct constructions for different uses of the word "how." In terms of the verbal construction, both of the "what" constructions may correspond with S-lin-cxn ("What a good book she wrote!") or inf-cxn ("What a good book to read!"), but not S-inv-cxn ("*What a good book did she read"). However, both of the "how" constructions correspond with only S-lin-cxn. The "how" constructions paired with other types of verbal construction produce some of the other types of filler-gap constructions which will be explored in the following sections. In terms of inheritance relationships, all of the wh-exclamatory forms can stand on their own as sentences, so they can be said to inherit from S. Finally, note that, in order for the filler and gap to match in syntactic and semantic category, the DP fillers take a gap in the complement, while the AdjP and AdvP fillers take a gap in the adjunct. The Wh-ex-cxn can now be defined:

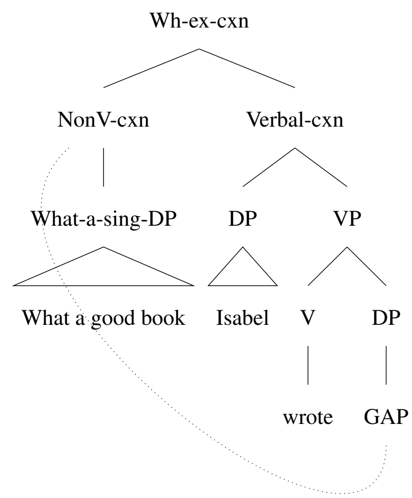
Definition 4.4. *The Wh-Exclamative construction may be defined as the set of all elementary initial trees in FG-cxn-init whose filler is in one of the sets What-a-sing-DP or What-pl-DP and whose main clause is in one of the sets S-lin-cxn or inf-cxn, or whose filler is in one of the sets How-AdjP or How-AdvP and whose main clause is in the set S-lin-cxn.*

Of course, there is a large number of elementary trees which meet these specifications to qualify as a wh-exclamative but some are displayed below. Figure 16 contains a What-a-DP in the filler, a S-lin-cxn as the main clause, and a DP complement as the gap. Figure 17 contains a How-AdvP in the filler, a S-lin-cxn as the main clause, and an AdvP adjunct as the gap. Figures 18 and 20 represent full lexicalizations of the initial trees in 17 and 19 respectively.

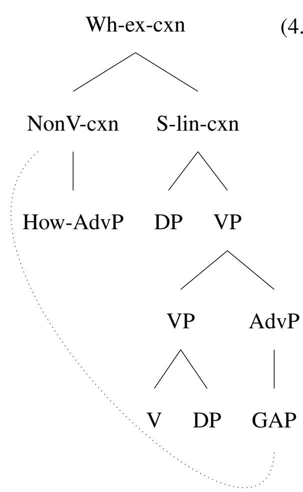
(4.16)



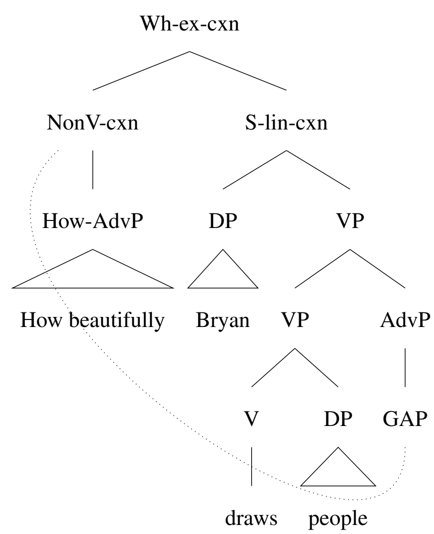
(4.18)



(4.17)



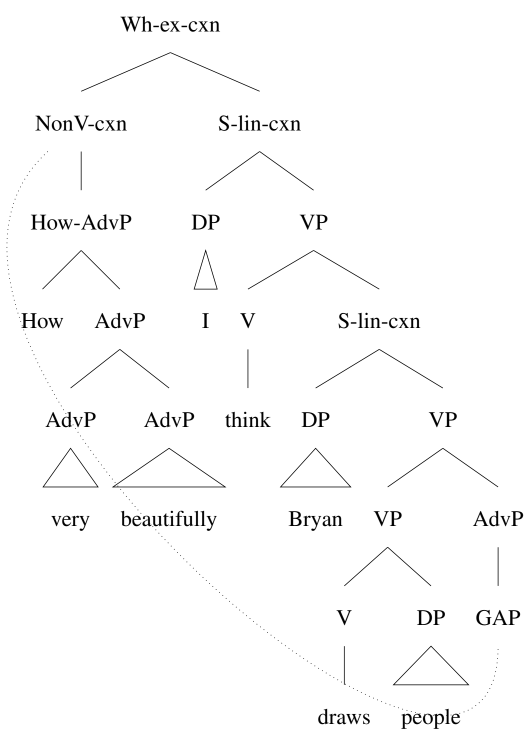
(4.19)



As the name suggests, the wh-exclamative construction defines the set of syntax trees whose semantic interpretation is an exclamation. For example, the sentence “Isabel wrote a good book” may be interpreted as an exclamation or simply as an indicative statement, based on punctuation marks in writing or tone in speech. Conversely, “What a good book Isabel wrote(!)” indicates an exclamation by virtue of the construction, without necessarily needing to indicate that semantic interpretation with a punctuation mark or tone. Furthermore, wh-exclamatives often sound much better with some sort of degree word in the filler to grant emphasis to the exclamation. For example, ”What a book Isabel wrote!” is acceptable but less pleasing than ”What a good book Isabel wrote!” Again, we can see that the construction itself contains some element of meaning above and beyond just the meanings of the lexical items inserted into it.

Just as with the topicalized clauses, the constructions themselves are acquired as lexical items belonging to a certain category and must be individually specified as initial trees; however they may expanded recursively using auxiliary trees. The following example demonstrates modification of the simple Wh-ex-cxn initial trees with auxiliary trees for adjectives and adverbs that can be adjoined at the NP and VP nodes anywhere on the initial trees.

(4.20)



-

4.5 Wh-Interrogatives

The wh-interrogative is another construction which makes use of wh-constructions, in fact far more than the wh-exclamative. The wh-interrogative filler may accept What-pl-DP, How-AdjP, and How-AdvP (but not What-a-sing-DP), demonstrated in the previous subsection. It may also accept Who-DP (21), Whose-DP (22), which-DP(23), What-sing-DP (24), When-PP (25), Where-PP (26), and Why-PP (27).

(4.21) **Who** did you see?

(4.22) **Whose mother** did you see?

(4.23) **Which city** did you visit?

(4.24) **What book** did you read?

(4.25) **When** did you leave?

(4.26) **Where** did you go?

(4.27) **Why** did you return?

Note that the Wh-prepositional phrases do not contain stated prepositions but function syntactically as prepositional phrases. For example, "When did they arrive?" corresponds to "They arrived in the morning" which is a temporal prepositional phrase. If a wh-PP is used, the link with the gap means that if the wh-PP is the temporal When-PP, then the gap must also be able to accept a temporal PP such as "in the morning." If the wh-PP is the spatial Where-PP, then the gap must be able to accept a spatial PP such as "in the kitchen." If the wh-PP is the causative Why-PP, then the gap must be able to accept a causative PP such as"because I said so." The case of the wh-PPs highlights the stipulation in the definition of FG-cxn-init that the filler and gap must match in semantic category. If this condition is not met, ungrammatical examples such as (28) may be produced in addition to grammatical examples such as (29).

(4.28) *When are the dogs?

(4.29) Where are the dogs?

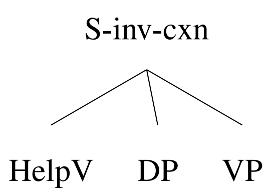
This study will break down wh-interrogatives into two separate constructions. One will be the wh-question Wh-q, and the other will be the wh-interrogative dependent clause Wh-int-dep. Both types can accept all of the wh-constructions listed above as the filler. The difference between these two lies in the main clause. Consider the following examples

(4.30) I know what you will eat.

(4.31) What will you eat?

In the Wh-q (30), the main clause "will you eat" is inverted, meaning that the helping verb precedes the subject. This is an example of the S-inv-cxn, or inverted sentence, which is the only type of Verbal-cxn which a wh-q can accept. The S-inv-cxn is composed of a helping verb, a DP containing the subject, and the main VP.

(4.32)



Since verb morphology is beyond the scope of this paper, the agreement between the helping verb and the main verb will not be addressed. One final point to address before providing the definition of the Wh-q is that there must be a constraint on the S-inv-cxn node specifying that no auxiliary trees with foot S* may be adjoined at that node. Consider the following examples:

(4.33) Who will Liz call?

(4.34) *Who Robin says will Liz call?

(4.35) Who does Robin say Liz will call?

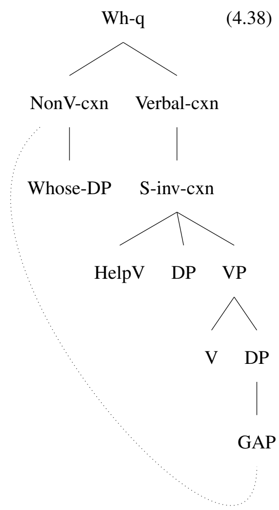
From these examples, it is evident that the auxiliary tree for ”Robin says” cannot be adjoined as in (34) once (33) has been constructed. Instead, (35) builds the inverted sentence with ”does Robin say” and substitutes ”Liz will call GAP” as the complement.

The Wh-q construction is defined as follows:

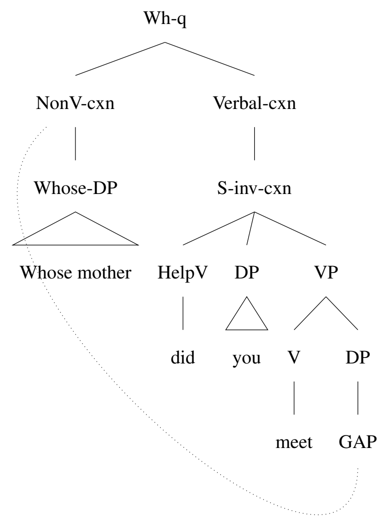
Definition 4.5. *The Wh-q construction is defined as the set of all elementary initial trees in FG-cxn-init whose filler is in one of the sets What-sing-DP, What-pl-DP, How-AdjP, How-AdvP, Wh-NP, Whose-DP, When-PP, Where-PP, or Why-PP, and whose main clause is in S-inv-cxn. No auxiliary tree with foot S* may be adjoined to the main verb of a Wh-q.*

Some examples of initial trees which belong to the wh-q are shown below. Figure 36 has a Whose-DP as the filler, an S-inv-cxn as the main clause, and a complement DP as the gap. Figure 37 has a Where-PP as the filler, a S-inv-cxn as the main clause, and an adjunct PP as the gap. Figures 38 and 39 are full lexicalizations of Figures 36 and 37. Particularly important is the fact that gap remains an adjunct to the main verb ”study” rather than the helping verb ”will.”

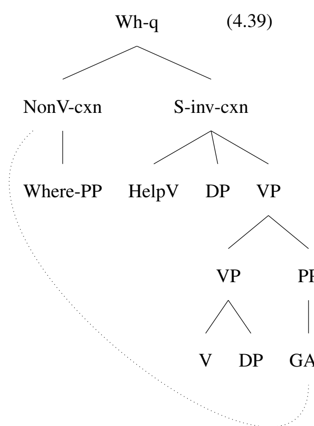
(4.36)



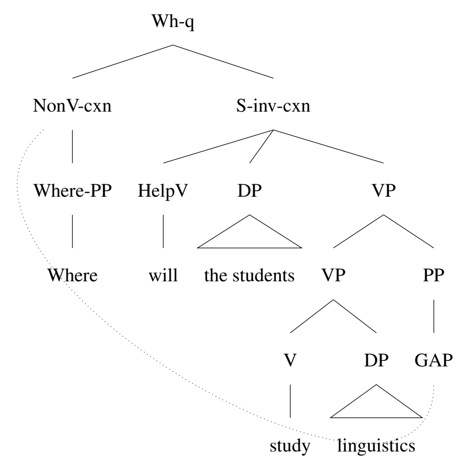
(4.38)



(4.37)



(4.39)



The other type of wh-interrogative is Wh-int-dep, the interrogative dependent clause. The Wh-int-dep may accept all of the same fillers as the Wh-q. The primary differences between the two are 1. that the Wh-int-dep accepts S-lin-cxn and inf-cxn as the main clause and 2. Wh-int-dep does not inherit from S. That is, any tree in the set Wh-int-dep cannot be fully lexicalized into a grammatical sentence. It must exist as the complement to a verb.

(4.40) She asked me what the doctor said.

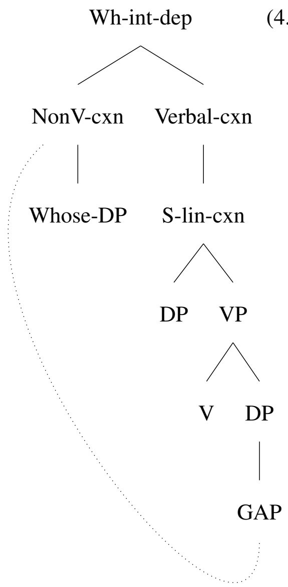
(4.41) I know what to do.

(4.42) *She asked me what did the doctor say.

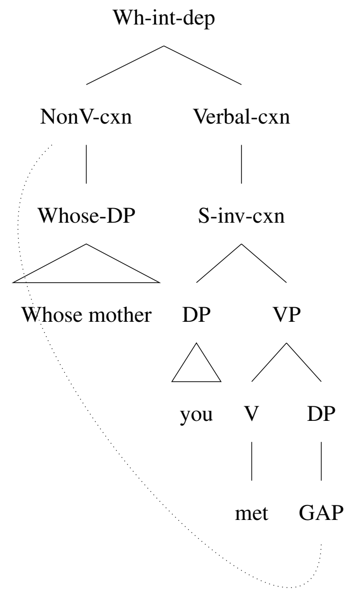
(4.43) *What the doctor said.

Definition 4.6. *The Wh-int-dep construction is defined as the set of all elementary initial trees in FG-cxn-init whose filler is in one of the sets What-sing-DP, What-pl-DP, How-AdjP, How-AdvP, Wh-NP, Whose-DP, When-PP, Where-PP, or Why-PP, and whose main clause is in S-lin-cxn or S-inf-cxn.*

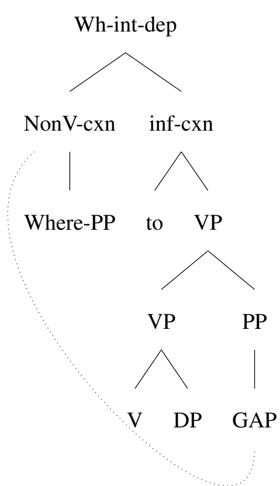
(4.44)



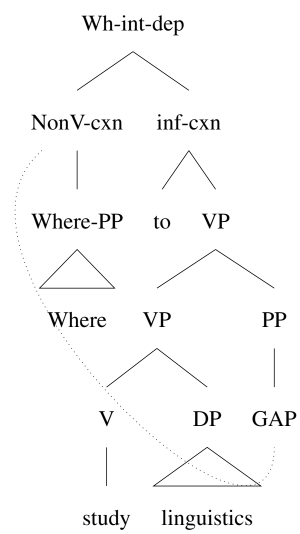
(4.46)



(4.45)



(4.47)



These examples demonstrate the difference between the dependent interrogative and the wh-question– the inverted sentences in the verbal construction are replaced with linear sentences and infinitives in examples 44 and 45 respectively. The full lexicalizations shown in 46 and 47 demonstrate that dependent interrogatives cannot stand on their own, but rather must be substituted into an S construction such as ”I know whose mother you met” or ”Ask her where to study linguistics.”

In terms of semantics, the wh-interrogative constructions share semantic meaning from the construction regardless of which lexical items are substituted into the construction. As the name of the construction suggests, both the Wh-q and Wh-int-dep interrogatives have a question sense– Wh-q is a question itself, and Wh-int-dep occurs as the complement of either a questioning word (wonder, ask), or an answering word (know, believe etc). It is also interesting to note that the Wh-q accepts a S-inv-cxn as the main verb and carries the semantic sense of the direct question itself, while Wh-int-dep does not accept S-inv-cxn as the main verb and carries the semantics of an indirect question. This is interesting because non-wh questions also have the S-inv-cxn as the main clause, and indirect statements and indirect commands accept S-lin-cxn and inf-cxn respectively.

4.6 Wh-Relatives

The final type of FG-cxn that will be addressed in this paper is the wh-relative construction, or Wh-rel-cxn. Relative clauses are dependent clauses; that is, they do not stand on their own as sentences and instead function as modifiers. They contain a relative pronoun such as who, whom, whose, which, or that, that refers back to whatever DP the relative clause is modifying, and also a verb phrase. However, not all relative clauses belong to FG-cxn. Consider the following examples:

(4.48) I know the person **who you know**.

(4.49) I know the person **who knows you**.

(48) is a filler-gap construction because ”know” has an empty valency and ”who” has the same syntactic and semantic categories as the gap in the VP. (49) is not a filler-gap construction because ”who” is the subject of ”knows,” and ”you” is the object of knows, so ”knows” has no empty valencies. ”Who” is also not any type of adjunct that could adjoin to ”knows,” so it cannot be a filler to an adjunct gap in the VP either. To differentiate the filler-gap relatives from other relatives, we will call the filler-gap relative FG-rel-cxn. The FG-rel-cxn contains every initial tree that belongs to one of the wh-nonverbal constructions as follows: Who-DP, Whose-DP, Which-DP, That-DP,

When-PP, Where-PP, Why-PP. The relative clause does not accept ”what” or ”how” constructions. The main clause can be either S-lin-cxn or inf-cxn, but the inf-cxn must have a prepositional phrase as the filler.

(4.50) The person **whose child I babysat** returned home.

(4.51) *The essay **what I wrote** was good.

(4.52) *The method **how I solved the problem** was complicated.

(4.53) The best place **in which to stay** is Quincy House.

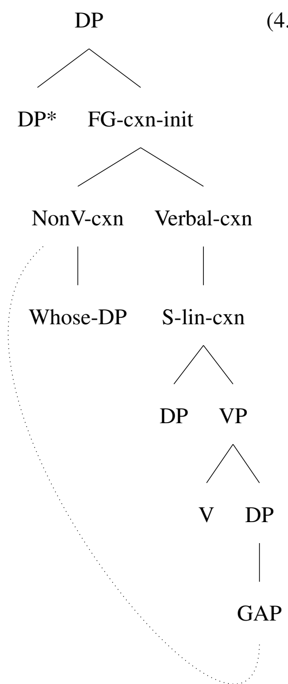
(4.54) *The best place **where to stay** is Quincy House.

Finally, it is important to note that, since relative clauses modify DP, they are adjuncts and therefore must inherit from FG-cxn-aux rather than FC-cxn-init as the other three cases have.

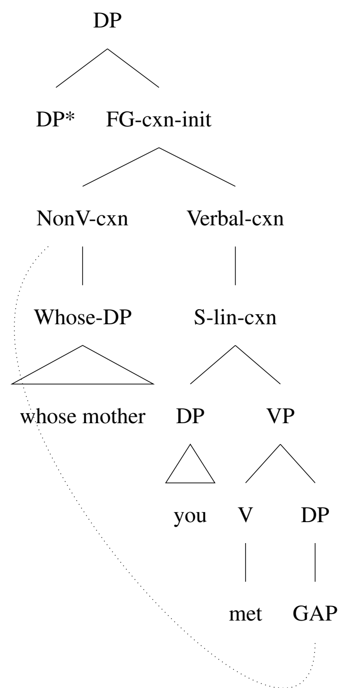
Definition 4.7. *The FG-rel-cxn construction is defined as the set of all elementary auxiliary trees in FG-cxn-aux whose filler is in one of the sets Who-DP, Whose-DP, Which-DP, That-DP, When-PP, Where-PP, Why-PP, and whose main clause is in S-lin-cxn, or whose filler is in When-PP, Where-PP, or Why-PP and whose gap is in inf-cxn.*

One unique property of the wh-relative vs the other types of filler-gap constructions is that FG-rel-cxn serves as an adjunct and therefore must be adjoinable at the correct nodes in the TAG representation, while the others either are substituted as a complement or stand alone as their own sentences. It is because of this that wh-relatives can be “stacked” unlike the other types of FG-cxn. Consider the following examples:

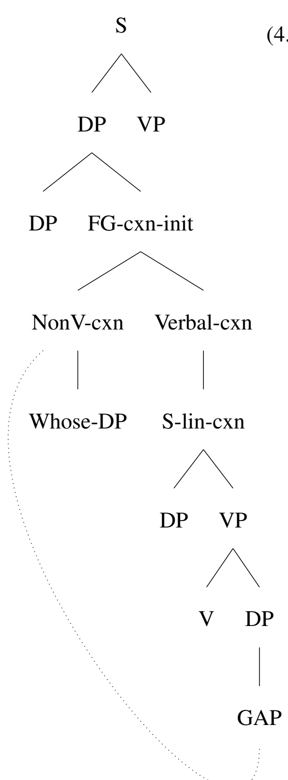
(4.55)



(4.57)



(4.56)



(4.58)

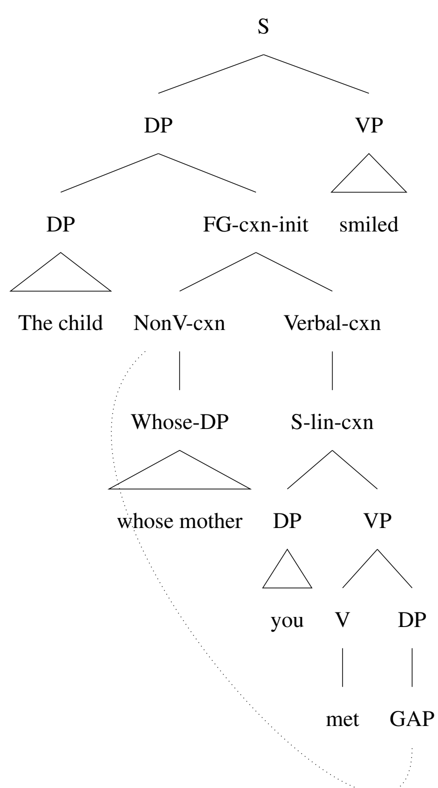


Figure 55 represents an auxiliary tree in the set FG-rel-cxn which has a Whose-DP as the nonverbal construction and a linear sentence as the verbal construction. Figure 56 represents adjoining that auxiliary tree to the simple S construction composed of DP and VP. Figure 57 represents a full lexicalization of Figure 55, and Figure 58 represents a full lexicalization of Figure 56.

This concludes the description of the four selected filler-gap constructions using the TAG approach. There are, of course, other less commonly occuring filler-gap constructions such as the ”the-clause” (”The taller you are, the better you play basketball”) and as-fronting (”as good as it gets”). It is apparent that the limitations of this study must be addressed. What follows, now, is an assessment of how well this TAG-Construction Grammar approach to filler-gaps addresses the issues with the wh-movement approach.

5 Analysis and Discussion

5.1 Introduction

The initial objective of this paper was to assess a Construction Grammar approach to filler-gap sentences using a TAG implementation in order to determine whether such an approach offers any advantages over the standard wh-movement paradigm. Two primary flaws of wh-movement were noted for possible improvement: 1. Wh-movement tends to overgenerate ungrammatical sentences, unless restrained by several additional rules. Since many specific constraints must be introduced to sufficiently limit the production of ungrammatical sentences, from a computational perspective, it eliminates the advantages of having a generalized move operation since the specific contexts in which the operation may be used must be encoded as well and 2. The types of structures produced by wh-movement can have vastly different semantic values, which cannot be accounted for using only the definitions of the individual words; the syntax contributes strongly to the semantic interpretation as well. This chapter will discuss how successful the TAG CxG implementation was in alleviating these primary flaws with wh-movement. Furthermore, it will address additional flaws which the CxG approach introduces and proposes how they may be resolved in future studies. Finally, it will describe some empirical methods of comparing the efficacy of the CxG approach and the wh-movement approach for engineering applications that may be implemented in future studies.

5.2 Construction Grammar Approach Eliminates Most Overgeneration

The CxG approach to filler-gap sentences seems to eliminate common instances of overgeneration in wh-movement which must be ruled out by additional restrictions. The wh-movement paradigm often overgenerates without additional restrictions when the category of the wh-moved element is permissible in some instances, but not permissible in others. Consider the following examples.

(5.1) I wonder who she likes.

(5.2) I wonder **how fast she is**.

(5.3) The thing **which she did**...

(5.4) *The speed **how fast she is**...

(5.5) *The thing **what she did**...

The ungrammatical sentences in examples 4 and 5 can never occur in the Construction Grammar approach. The only filler-gaps which can modify DPs such as ”the speed” or ”the thing” are those in FG-rel-cxn, which does not accept ”how” or ”what” wh-constructions in the filler. Conversely, wh-movement allows the AdvP to move from the adjunct of the main verb into spec-CP to create the form in example 2 ”how fast she is.” That exact same movement is not permissible in example 4. Additional rules must apply to restrict the movement of the AdvP to spec-CP of the complement of the main verb, and not the subject. Likewise, wh-movement allows a DP to move from the complement of the VP into spec-CP to create example 3 ”which she did.” That same movement is not permissible in example 5, requiring an additional rule about which determiners can be moved. What these two examples highlight is that the lexical elements and syntactic types which can be moved, as well as the locations of the specifiers into which they may be moved, are far more specific than a simple DP to spec-CP or AdvP to spec-CP move operation. In essence, the wh-movement approach defines a highly broad rule and provides several exceptions and restrictions. On the other hand, Construction Grammar requires a great deal more specificity to begin with, but ultimately its net generality is quite similar to the wh-movement approach after all the exceptions and restrictions.

Consider another set of examples:

(5.6) He will study yoga.

(5.7) **Yoga**, he will study.

(5.8) Will she study yoga?

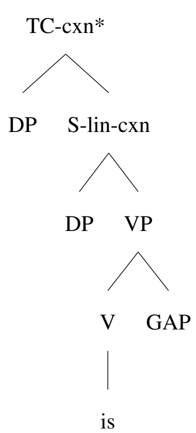
(5.9) *$\mathbf{Yoga}$, will she study?

(5.10) She is healthy.

(5.11) **Healthy**, she may be.

(5.12) ***Healthy**, she is.

The Construction Grammar approach would not be able to produce the ungrammatical sentences (9) because the only filler-gap which does not have a wh-word is the topicalized clause, and the topicalized clause must have a linear sentence as the verbal construction; it cannot be inverted as is the case in sentence (9). Construction Grammar also provides an easy way to eliminate a sentence like (12), where topicalizing with the verb ”is” sounds strange. When designating the original set $G = (I^*, A^*)$, simply do not include any elementary trees anchored in ”is” which meet the requirements for a topicalized clause, as such:



Sentences (9) and (12) create issues for the wh-movement paradigm, because we can see from example (7) that a DP complement can move into spec-CP. Additional restrictions are necessary to ensure that the DP complement does not move into spec-CP when the main VP is inverted, and to ensure that the DP complement does not move when the main verb is ”is” (but that restriction does not apply to other conjugations of the word ”is” such as the subjunctive ”may be”). What this set of examples demonstrates is that the type or even the specific conjugation of the main verb, and associated semantic effects such as the interrogative (9) or subjunctive (11), can restrict the application of the move-operation further.

5.3 Eliminating Island Effects

One well-studied category of limitations on the move operation which this chapter will highlight specifically is the ”wh-island,” or the set of contexts from which no wh-movement is possible. This section will demonstrate how the Construction Grammar cannot produce any ungrammatical utterances which may be produced if a node is moved out of a wh-island. Consider the following examples:

(5.13) I met Greg’s mother.

(5.14) *Whose I met mother.

(5.15) She is so funny.

(5.16) *How she is funny.

These examples above demonstrate that a modifier to the complement of the main verb which appears before/to the left of the complement itself constitutes a wh-island. The move operation does not apply in any such situation. The CxG-TAG approach also avoids producing any examples such as (14) and (16). This is because the Whose-DP and How-AdvP require complements. That is to say, ”Whose mother” is a Whose-DP, but ”Whose” is not because there is an unsubstituted node in the DP. Likewise, ”How funny” is a How-AdvP, but ”How” is not, because there is an unsubstituted node in the AdvP. Furthermore, in order to maintain the syntactic and semantic equivalence between the filler and the gap, if the filler is a How-AdvP, the gap must be an AdvP, not an Adv, and so forth.

To examine another type of wh-island, consider the following examples:

(5.17) I will go to the party because I finished my homework.

(5.18) *What will I go to the party because I finished?

This is another example of an island effect. Wh-movement is not permissible outside of subjunctive clauses such as those headed by ”if,” ”when,” ”because” etc. The CxG approach will not produce such a sentence because the gap is the complement of an adjunct to the main verb “go.” The gap must contain the entire adjunct to the main verb of an S construction, and the filler must match it in type, so only sentences such as “Why will I go to the party?” will be accepted. Note, it also will not accept “Because I finish my homework will I go to the party?” because “Because I finished my homework” is not a wh-construction appropriate for the filler of a Wh-q.

These examples illustrate another island effect:

(5.19) What did you give him?

(5.20) Why did you give him that?

(5.21) What did she know you gave him?

(5.22) *What did she know why you gave him?

This is island restricts wh-movement such that, nothing can be moved outside of a clause which has already undergone a wh-movement transformation. The CxG approach will not generate sentence (22), since the gap is in the complement of “give,” which is not the main verb of an S construction. Rather, is the main verb of the embedded Wh-int-dep “why you gave him GAP” which is itself the complement of the main verb “know.” The gap must be the entire complement or adjunct of the main verb of an S construction, not simply a part of an incomplete adjunct or complement.

Here is a final set of examples illustrating island effects:

(5.23) She thinks he went to the zoo.

(5.24) Where does she think he went?

(5.25) She wonders who went to the zoo.

(5.26) *Where does she wonder who went?

(5.27) She wonders when Hannah saw Davis.

(5.28) *Who did she wonder when Hannah saw?

This final type of island is observed when attempting to move an element out of a wh-clause as demonstrated by examples (26) and (28). Moving the spatial (prepositional) phrase into the higher spec-CP rather than the lower spec-CP occupied by the other wh-word (in this case ”who”) is not permissible. The Wh-q construction neatly deals with this case. Sentence (24) is permissible because the gap occurs in the adjunct of a VP contained in an S, since ”he went GAP” is a linear sentence. Sentence (26) is not permissible because the gap occurs in the adjunct of a VP in a Wh-rel-cxn ”who went GAP.” Sentence (28) is not permissible because the gap occurs in the complement of a VP in Wh-int-dep. As established in Chapter 4, Wh-rel-cxn and Wh-int-dep cannot stand on their own as individual sentences therefore the complements and adjuncts to the verbs they contain are not satisfactory gap positions.

In all of these situations, the Construction Grammar approach is able to simply apply the rules of adjunction and substitution to a highly specific but predetermined set of elementary trees, and common ”island effects” are not observed. The wh-movement paradigm, of course, requires a detailed set of restrictions on wh-movement to prevent movement out of islands.

5.4 Syntactic Advantages and Limitations

What all of these examples are showing is that while general move operations may be defined for a broad variety of syntactic structures, specific situational rules are required to limit when the move operation may be used in order to avoid overgeneration. Construction Grammar has been criticized in comparison to competing theories because it posits that far more syntactic patterns are memorized rather than generated from a set of recursive rules. In other words, rather than having one deep structure that may be transformed with move operations in different ways to create several different surface structures with closely related meanings, Construction Grammar posits that each of the variant surface structures is uniquely acquired. However, the above examples demonstrate that, when accounting for the restriction rules on the move operation as well as the operation itself, the CxG and wh-movement approaches ultimately have a similar level of generalization (or lack thereof). There is a finite set of structures which can be "moved" and a finite set of contexts from which each of those structures may be "moved," and a person must still learn/memorize each correct combination of origin and destination. That is to say, the move operation is not infinitely generative the way chained AdvPs or PPs are. Therefore, the CxG-TAG approach is successful because it encodes both the finite and infinite aspects of language. There is a finite number of initial trees in the set FG-cxn-init, representing the finite number of combinations of filler type, main clause type, and gap type. However, there are infinitely many speech utterances that may be produced by recursively applying auxiliary trees such as FG-rel-cxn or even AdvP and PP. The CxG-TAG approach is a better choice than the wh-movement approach when generating text artificially in situations where accuracy is paramount. Since the lexicon of FG-cxn constructions from which the text would be generated is specifically built to ensure that only correct combinations of filler, main clause, and gap type are registered as constructions, no overgeneration would occur.

However, it is critical to observe some very evident limitations of this study. Firstly, filler-gaps were selected for this study because they had been previously identified as a set of syntactic structures which would work well with Construction Grammar, and which have historically worked somewhat poorly with other approaches. This study was able to provide some substantiation for those claims, as the CxG-TAG approach handles many difficult filler-gap constructions using only the adjunction and substitution operations. However, a true assessment of Construction Grammar in comparison to more mainstream grammars would be to provide formalization for syntactic structures on which UG performs exceptionally well, and eventually for all syntactic structures. Just because Construction Grammar handles an extremely small set of syntactic structures well does not mean it will perform as well as other approaches in any other context.

A second limitation of this study is that it took a primarily theoretical approach. Although theory is critical for developing empirical studies, it seems premature to claim that Construction Grammar performs better than wh-movement on filler-gaps based solely on theory without experimental data to support the claim. This introduces the third and largest limitation of the approach presented in this paper. While Construction Grammar simplifies the processes of parsing and generating text by eliminating rule-exceptions, it monumentally adds to the task of actually defining the language. This study focused simply on four constructions, but millions, at the very least, would be necessary to define the English language. Without the complete set of English constructions, text generation and parsing would be prone to undergeneration. That is, if the constructions for a particular structure have been left out of the set of elementary trees, the machine would not be able to parse or generate any instances of those structures in its attempt to eliminate all overgeneration. Sophisticated computational methods such as those used for part-of-speech tagging would almost certainly be necessary to generate the finite (but very large) set of elementary constructions. For many potential experiments aiming to compare Construction Grammar based parsing or text generation to existing parsers and AIs, creating the set of constructions defining the language would be a very difficult first step.

5.5 Semantic Advantages and Limitations

This study has primarily focused on developing a model that generates and identifies correct syntactic structures. However, there are certain semantic advantages to CxG which appear in the subset of constructions analyzed in this paper. One of the fundamental hypotheses of CxG is that constructions, rather than words, are the lexemes of a language. Therefore, every construction has meaning structurally regardless of what words are substituted into it. The model developed in this paper provides evidence to support this hypothesis. As discussed briefly in the methods section, each individual filler gap construction carries a unique semantic interpretation which is different from the basic indicative S construction, and also different from the other FG-cxns. The topicalized clause implies contrast, and often sounds strange without a second, contrasting topicalized clause. The wh-exclamative implies an exclamation, and therefore often sounds better with a degree word. The wh-interrogative is a direct or indirect question, and the indirect question sounds strange in contexts other than after a verb of questioning. Additionally, the Wh-q and the basic question with no wh-element both feature inverted verbal constructions:

(5.29) What food did Aaron cook?

(5.30) Did Aaron cook food?

This lends evidence to the fact that the form of an inverted verbal construction offers it some interrogative meaning without the addition of any specific lexical elements. Finally, the wh-relative must serve as a description of another nominal lexeme.

All the variations of the filler-gap clause can be uniquely determined by the composition of its construction– the type of the filler, the main clause, and the overall construction. Each of the four variations also has a unique meaning, since changing the structure even slightly can shift the interpretation. This would seem to be evidence to support the fact that the construction has meaning itself, regardless of the individual words. Consider "Wugs, I will wump." Even with the nonsense words, there is the underlying implication of "but something else, I won't wump." The semantic value of contrast must be coming from the syntax, since it cannot be coming from the meaning of the nonsense noun and verb used. The fact that the filler is a noun with no wh-word and the main clause is a linear sentence means that this sentence is a TC-cxn, and that means that it must have some semantic element of contrast.

This semantic value encoded in the constructions could potentially be another advantage of CxG over the move operation. The move operation transforms a "deep structure" into the surface structure of the actual sentence, which demonstrates the semantic relationship between the two. For example, parsing "What did she say?" as a manipulation of "She said that" shows the semantic relationship between these two sentences. However, the same generalized move operation is used across all types of filler-gap sentences, as well as for some other types of sentences. Therefore, this generic syntactic operation cannot account for the unique meaning of all different filler-gap constructions which use it. As previously established, the individual vocabulary words cannot fully explain the different semantic interpretations as well. As a result, it seems that the wh-movement explanation fails to explain some of the inherent semantic differences between the filler-gap variations.

The semantics of filler-gap constructions could be an interesting area for further research. Since this study was focused on syntax, no frame semantics were used to restrict which nouns could be used as complements of specific verbs, etc. That is, the paradigm described above follows "colorless green ideas sleep furiously" syntax, in that it will accept any words which fall into the correct syntactic categories without regard for whether or not those words combine meaningfully. Although it is informally specified, for example, that "whose ideas did I sleep?" does not satisfy the requirements for Wh-q in Chapter 4, since the filler must match the gap semantically as well as syntactically, no formal mechanism of assessing this match is given at this time. Therefore, a potential next step would be to develop a way to encode frame semantics, a popular method of semantic analysis for construction grammarians, into the TAG representation of construction grammar.

In summation, the study presented here provides a theoretical justification to substantiate some of the fundamental principles of Construction Grammar as they apply to filler-gap variations. However, this study is quite limited in scope, in terms of both the small number of constructions it examines and the theoretical methods it employs. In addition, this study has also highlighted some weaknesses of the Construction Grammar approach, especially the sheer magnitude of the set of elementary trees. This criticism aligns with the usual criticism that Construction Grammar receives. While concluding that Construction Grammar is or is not superior to other grammars, in the realm of filler gaps or otherwise, would be hasty, it can be concluded that this avenue of research is at least worthy of further study.

6 Conclusion

At the beginning of this paper, Construction Grammar was identified as a potential alternative theory of grammar which may compare favorably to more established theories. Filler-gap constructions were identified as an area of syntax where Construction Grammar might perform exceptionally well, and the objective, therefore, was to assess from a theoretical standpoint how CxG compares to UG. In order to make a proper theoretical assessment of CxG, the principles of CxG needed to be translated into a mathematical formalism. Over the course of this paper, the principles of CxG were defined using the Tree-Adjoining Grammar formalism, and the four variations of filler-gap were defined using that TAG representation. As was predicted, the Construction Grammar approach successfully avoided overgeneration and accounted for semantic differences between the different filler-gaps. However, several weaknesses in terms of both the scope of the study and the extreme specificity of Construction Grammar itself were raised. This final chapter will expand upon some potential avenues of future research to address the weaknesses of the study.

To first address the limitations of the scope of the study, the primary avenue of future research would be to develop an automated system to generate the set of elementary trees. As mentioned in Chapter 5, the set of all elementary trees necessary to define a language is far too large to define by hand the way this paper defines a very small subset of elementary trees. A machine learning approach would probably be best suited to addressing this problem. Given sample speech as training data and a subset of elementary trees as a starting point, the algorithm should be able to identify speech utterances which do not fit into any of the constructions it currently knows and should then predict the composition of the new construction based on other constructions it has previously seen and the context in which it appears. Each newly developed construction would then be assigned a score based on how often it can be applied to speech utterances the machine encounters later. If the new construction is immediately used over and over, it is likely that this construction does exist in the language and therefore it receives a high score. Conversely, if that construction is never instantiated again there is a chance a mistake has been made in creating that construction, and it is assigned a low score. A good example of a study which uses this method to identify constructions is Luc Steels's 2011 paper on his formalism, Fluid Construction Grammar. Once all of the elementary trees (or at least, a large enough subset of them) have been identified, the process of experimenting with Construction Grammar in comparison to other grammars becomes much simpler. Furthermore, the Construction Grammar is far less likely to under-generate if enough of the necessary constructions have been identified.

With a more complete set of elementary trees in place, the next step would be to expand the methods and scope of this current study to include experimental methods and other types of constructions. This paper was limited to filler-gap constructions, which had already been identified as a potential match with the principles of Construction Grammar and had numerous inconsistencies in the wh-movement approach. Furthermore, it was limited to theoretical analysis rather than comparison of empirical data. Future research should then include studies of many other types of grammatical constructions, perhaps identifying which structures work better with CxG and which work better with UG. A possible experiment would be to develop a parser that parses into TAG-based Construction Grammar, and compare its performance against any of the pre-existing parsers. From an engineering perspective, the value of a grammar correlates to how well it performs at comprehension and production tasks, so some sort of concrete measurement of the approach presented in this paper is necessary to determine how valuable it is. It may also be worthwhile from a linguistic and psychological standpoint to perform human-based experiments using this theory as well as computer-based experiments. While this study primarily focused on what utterances the filler-gap constructions can and cannot generate, a potential future study might examine the way Construction Grammar generates speech in comparison to how a human generates speech.

The final weakness identified in Chapter 5 was the lack of a formal integration of frame semantics into the TAG formalism. Since this study focused primarily on syntax, such an integration was elided. However, there does exist significant prior research on frame semantics in TAG, including the 2006 study by Nesson and Shieber on Synchronous TAG, which manipulates a syntax tree and a semantics tree simultaneously. A future theoretical study might more precisely define the semantic interpretation of the filler-gap constructions and how the syntax contributes to that interpretation by incorporating frame semantics into the existing TAG filler-gap constructions by introducing STAG.

Based on the limited, theoretical study presented in this paper, it is likely that a formalized Construction Grammar may present an improvement over wh-movement in the analysis of filler-gap sentences. Since this study is so limited, far more research must be done before drawing a conclusion on the merits of Construction Grammar as a whole. However, as demonstrated over the course of this chapter, there are many areas of future research which could shed light on this issue. Although no concrete evaluation of Construction Grammar may be provided at this time, it certainly merits further study.

7 References

Beuls, K. (2012). Grammatical error diagnosis in fluid construction grammar: a case study in L2 Spanish verb morphology. Computer Assisted Language Learning, 27(3), 246–260. doi: 10.1080/09588221.2012.724426

Blumstein, S. E., Byma, G., Kurowski, K., Hourihan, J., Brown, T., Hutchinson, A. (1998). On-Line Processing of Filler–Gap Constructions in Aphasia. Brain and Language, 61(2), 149–168. doi: 10.1006/brln.1997.1839

Ciortuz, L., Saveluc, V. (2012). Fluid Construction Grammar and Feature Constraint Logics. Computational Issues in Fluid Construction Grammar Lecture Notes in Computer Science, 289–311. doi: 10.1007/978-3-642-34120-$5_1$2

Croft, William. 2001. Radical Construction Grammar: Syntactic Theory in Typological Perspective. New York: Oxford University Press.

Goldberg, Adele E. 2013. ”Constructionist Approaches.” In The Oxford Handbook of Construction Grammar, edited by Thomas Hoffmann and Graeme Trousdale.

Hoffmann, Thomas. 2013. ”Abstract Phrasal and Clausal Constructions.” In The Oxford Handbook of Construction Grammar, edited by Thomas Hoffmann and Graeme Trousdale. DOI 10.1093/oxfordhb/9780195396683.013.0002.

Kroch, Anthony and Joshi, Aravind. 1985. ”The Linguistic Relevance of Tree Adjoining Grammar.” University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-85-16.

Lichte, Timm and Kallmeyer, Laura. ”Tree-Adjoining Grammar: A Tree-Based Constructionist Grammar Framework for Natural Language Understanding.” In AAAI 2017 Spring Symposium on Computational Construction Grammar and Natural Language (Palo Alto, CA, 2017)

Nesson Rebecca and Shieber, Stuart M. ”Simpler TAG semantics through synchronization.” In Proceedings of the 11th Conference on Formal Grammar (Malaga, Spain, 29-30 July 2006). Sag, Ivan A. 2010. ”English Filler-gap constructions.” Language 86, no 3: 486-545.

University of Pennsylvania. (2020). A Lexicalized tree adjoining grammar for English. Philadelphia, PA: University of Pennsylvania, School of Engineering and Applied Science, Dept. of Computer and Information Science.

Steels, L. (2011). A design pattern for phrasal constructions. Constructional Approaches to Language Design Patterns in Fluid Construction Grammar, 71–114. doi: 10.1075/cal.11.06ste.

Storoshenko, D. R., Han, C.-H. (2013). Using synchronous tree adjoining grammar to model the typology of bound variable pronouns. Journal of Logic and Computation, 25(2), 371–403. doi: 10.1093/logcom/exs064

Trijp, R. V. (2015). Cognitive vs. generative construction grammar: The case of coercion and argument structure. Cognitive Linguistics, 26(4). doi: 10.1515/cog-2014-0074

Trijp, R. V. (2019). How a Construction Grammar account solves the auxiliary controversy. Benjamins Current Topics Case Studies in Fluid Construction Grammar, 79–104.

Vijay-Shanker, K. 1992. ”Using descriptions of trees in a tree adjoining grammar.” Computational Linguistics 18 no. 4: 481-517. https://www.aclweb.org/anthology/J92- 4004