Speech and Language Processing

by Jurafsky, Martin

12 Constituency Grammars

12.1 Constituency

Syntactic constituency is the notion of groups of words behaving as single units (constituents).
Evidence that a group of words forms a constituent can be that

- they occur in similar syntactic environments
- they cannot be split up, even if the group itself can be moved (preposed or postposed)

12.2 Context-Free Grammars

CFGs (or PSGs) are the most widely used formal system for modelling constituent structure in natural languages. The formalism is equivalent to the Backus-Naur Form (BNF).

A CFG consists of a set of rules (or productions), and a lexicon. For example,
$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
Nominal $\rightarrow$ Noun | Nominal Noun

The symbols in a CFG are either terminals or non-terminals (variables). The non-terminal associated with each item in the lexicon is its part of speech (lexical category).

A CFG is said to generate the set of strings that can be derived from its rules from a designated start symbol. It is common to represent derivations by parse trees. Thus, a CFG defines a formal language consisting of grammatical sentences.

This kind of use of formal languages to model natural languages is called generative grammar.

12.2.1 Formal Definition of Context-Free Grammar

A CFG G is defined by a 4-tuple (N, Σ, R, S):

- N is a set of non-terminal symbols (variables)
- Σ is a set of terminal symbols ($\Sigma \cap N = \phi$)
- R is a set of rules
- S is a designated start symbol ($S \in N$)

Direct derivation is a one-step substitution by a rule. Derivation is a straightforward generalisation of this.

12.3 Some Grammar Rules for English

12.3.1 Sentence-Level Constructions

Four constructions are common: declaratives, imperatives, yes-no questions and wh-questions. There are two types of wh-questions: wh-subject-questions and wh-non-subject-questions. The latter construction involves a long-distance dependency; for example, the relationship between `flights` and `have` in *What flights do you have from A to B?*.

12.3.2 Clauses and Sentences

A clause is traditionally defined as a "complete thought". More rigorously, a clause is a node S below which the main verb has all its arguments.

12.3.3 The Noun Phrase

The grammar of L_0 above contained three types of noun phrases: pronouns, proper nouns and the `Det Nominal` construction. The latter kind of noun phrase consists of a head (the central noun) and various modifiers.

The Determiner The determiner can be an article, a demonstrative, a quantifier or a possessive.

The Nominal The nominal contains the pre- and post-head noun modifiers. The simplest kind of nominal consists of just a single noun.

Before the head noun, one finds word classes that occur after the determiner (post-determiners) – cardinals, ordinals, quantifiers and adjectives. Some quantifiers occur only with plural nouns.

After the head noun, there are classes of words called postmodifiers – prepositional phrases, non-finite clauses and relative clauses.

From these properties, we get some additional rules:
Det $\rightarrow NP's$
Nominal $\rightarrow$ Noun
Nominal $\rightarrow$ Nominal PP
Nominal $\rightarrow$ Nominal GerundVP
Nominal $\rightarrow$ Nominal RelClause
RelClause $\rightarrow$ who | that VP

Rules for gerund VPs are made in a way exactly similar to those for VPs.

Before the Noun Phrase Certain classes of words (predeterminers) modify NPs and occur before determiners, like *all*.

12.3.4 The Verb Phrase

Verb phrases can be followed by many types of constituents. If an entire embedded sentence forms such a constituent, it is called a sentential complement (like in *You said you had a 266-dollar fare.*)

Transitive verbs are those that take a direct object NP and intransitive ones are those that don't. While these two are traditional subcategories of verbs, modern grammars distinguish many more, based on the types of complements verbs can take – NPs, non-finite VPs, sentences, and so on.

The possible set of complements for a verb is called its subcategorisation frame.

It is possible to create rules for each type of complement, but this approach results in an increased number of rules and less generality.

$VP \rightarrow$ Verb-with-no-complement
$VP \rightarrow$ Verb-with-NP-comp NP
$VP \rightarrow$ Verb-with-S-comp S

12.3.5 Coordination

Almost any constituent can be conjoined by a conjunction like `and`, like NPs, VPs, and sentences. Some grammar formalisms represent this as a metarule like: $X \rightarrow X$ and X

12.4 Treebanks

A treebank is a corpus where each sentence is paired with a corresponding parse tree.

12.4.1 Example: The Penn Treebank Project

The Penn Treebank Project uses Lisp-like parenthetic notation to represent trees. Further, `-NONE-` is used as a syntactic trace for long-distance dependencies.

12.4.2 Treebanks as Grammars

The sentences in a treebank implicitly constitute a CFG. However, this often results in a large number of rules, which poses problems for probabilistic parsing algorithms.

12.4.3 Heads and Head Finding

Syntactic constituents can be associated with a lexical head. In one simple model, each rule is associated with a head, so all non-terminals are annotated with their heads.
Another approach is to identify heads dynamically depending on the tree, using an independent set of rules. For example, for an NP,

- If last word is tagged POS, use it.
- Else search for the first child (L to R) which is NN, NNP, NNPS, NX, POS or JJR.
- Else for the first child which is NP.
- Else for the first child which is $, ADJP or PRN.
- Else for the first child which is CD.
- Else for the first child which is JJ, JJS, RB or QP.
- Else last word.

12.5 Grammar Equivalence and Normal Form

A formal language is a set of strings. Two grammars are called (weakly) equivalent if they generate the same language; they are strongly equivalent if they assign the same parse tree to the same strings as well.

The Chomsky Normal Form (CNF) of a grammar is one in which the right-hand side of each rule has either two non-terminals or one terminal. Any grammar can be converted into a weakly equivalent grammar in CNF.

12.6 Lexicalised Grammars

These grammars minimise the role of the lexicon, but that makes them cumbersome. To overcome lexicon-related issues in grammar, numerous alternative approaches have been developed, like CCG, TAG, HPSG and LFG.

12.6.1 Combinatory Categorial Grammar (CCG)

Components The categorial approach consists of three major elements: a set of categories, a lexicon and a set of rules.

1. Categories
   Categories are defined inductively from atomic elements to single-argument functions from categories to categories. More formally,

   - $A \subseteq C$
   - $(X/Y), (X \setminus Y) \in C \forall X, Y \in C$

   (X/Y) is a function that takes type Y on its right and returns type X; $(X \setminus Y)$ is one that takes Y on its left and returns X.

   A, the set of atomic elements, is typically very small.

2. The Lexicon
   The lexicon is an assignment of categories to words. For example,
   flight : N
   Miami : NP
   cancel : $(S \setminus NP)/NP$

3. Rules
   The rules specify how functions and their arguments combine. The following

are common to all categorial grammars:
$X/YY \implies X$ (forward function application)
$YX \setminus Y \implies Y$ (backward function application)

Categorial grammar derivations are illustrated as growing down from the words.

We can handle coordination of constituents by the rule
X CONJ $X \implies X$

Operations So far, CCGs are equivalent to CFGs. However, we can extend its capabilities by adding operations on functions.

1. Composition
   We can combine $(X/Y)(Y/Z)$ to (X/Z) (forward composition) or $(Y \setminus Z)(X \setminus Y)$ to $(X \setminus Z)$ (backward composition).

2. Type Raising Type raising converts simple categories to functions; it takes a category and converts it to a function that needs as an argument a function that needs the original category (yes, twice). For example,
   $X \implies T/(T \setminus X)$
   $X \implies T \setminus (T/X)$
   Using type raising, the first NP of a sentence of the form $NP\ (S \setminus NP)/NP\ NP$ can be converted to $S/(S \setminus NP)$, which allows us to compose it forward and end up with S.
   This provides us with a L-to-R, word-by-word derivation that reflects the actual processing of language by humans. Also, it allows us to make use of non-constituent groups of words in conjunctions (p. 24).

Long-Distance Dependencies Consider the phrase *the flight that United diverted*. Here the second argument of *diverted* is *the flight*, which is not in its usual position to the right of the verb. We can solve this by type-raising *United* to $S/(S \setminus NP)$ (p. 25).

12.7 Summary

1. Groups of consecutive words often act as constituents, and can be modelled by CFGs/PSGs.
2. A CFG has a set of rules over a set of variables and terminals. Every CFG generates a set of strings, its CFL.
3. A generative grammar is any formal language that models the grammar of a natural language.
4. Sentence-level grammatical constructions can be modelled with CF rules.
5. An English NP can have determiners, numbers, quantifiers and adjective phrases before; and gerundive and infinite VPs after the head noun.
6. English subjects agree with the main verb in person and number.

7. Verbs can be subcategorised by their complements, like transitive vs intransitive.
8. Treebanks exist.
9. Any CFG can be converted to its CNF.
10. Lexicalised grammars place more emphasis on the lexicon.
11. CCG is important.