Lecture 1: Context-Free Grammar and related concepts

Tianpei Xie

Jun. 26th., 2022

Contents

1 Concepts and Terminology

The basics of linguistic study is the grammar and terminologies. We summarize a list of basic grammar terms. **Context-free grammars** (a.k.a. **Phrase-Structure Grammars**) are the backbone of many formal models of the syntax of natural language (and, for that matter, of computer languages). As such, they play a role in many computational applications, including grammar checking, semantic interpretation, dialogue understanding, and machine translation. They are powerful enough to express sophisticated relations among the words in a sentence, yet computationally tractable enough that efficient algorithms exist for parsing sentences with them [Jurafsky and Martin, 2014].

The basic terms we need in linguistic studies are listed below:

1. **syntax** (noun) and *syntactic*: the study of the ***rules*** for the *formation* of grammatical sentences in a language, or the study of the *patterns* of formation of sentences and phrases from words. the study of how words and morphemes combine to form larger units such as phrases and sentences.

2. **semantics** (noun) and *semantic*: the study of ***meaning***, or an *interpretation* of the meaning, of a word, sign, sentence, etc. The study of linguistic development by *classifying* and *examining* changes in meaning and form.

3. **lexicon** (noun) and *lexical*: the ***vocabulary*** of a *language*, an individual speaker or group of speakers, or a subject, or branch of knowledge (such as nautical or medical); In linguistics, a lexicon is a language's inventory of *lexemes*.

4. **grammar** and *grammatical*: a set of ***structural constraints*** on speakers' or writers' composition of clauses, phrases, and words. The term can also refer to the study of such constraints, a field that includes domains such as *phonology*, *morphology*, and *syntax*, often complemented by *phonetics*, *semantics*, and *pragmatics*.

5. Linguistic theories generally regard human languages as consisting of ***two*** parts: a ***lexicon***, essentially a catalogue of a language's words (its wordstock); and a ***grammar***, a system of rules which allow for the combination of those words into meaningful sentences. The lexicon is also thought to include bound morphemes, which cannot stand alone as words (such as most affixes).

6. **constituent** (noun): a word or a group of words that function as a **single unit** within a *hierarchical structure*. Many constituents are *phrases*. A **phrase** is a sequence of one or more words (in some theories two or more) built around a *head lexical item* and working as *a unit* within a sentence.

7. **phonology** (noun) and *phonological*: a branch of linguistics that studies how languages or dialects systematically organize their **sounds** (or constituent parts of signs, in sign languages). The term also refers to the sound or sign system of any particular language variety.

8. **morphology** (noun) and *morphological* : the study of words, how they are formed, and *their relationship to other words* in the same language. It analyzes the structure of words and parts of words such as **stems**, **root** words, **prefixes**, and **suffixes**. Morphology also looks at parts of speech, intonation and stress, and the ways context can change a word's pronunciation and meaning.

9. **phonetics** (noun) and *phonetic*: a branch of linguistics that studies how humans **produce** and **perceive sounds**, or in the case of sign languages, the equivalent aspects of sign.

10. **pragmatics** (noun) : the study of how ***context*** *contributes to meaning.* The field of study evaluates how human language is **utilized** in **social interactions**, as well as the relationship between the interpreter and the interpreted.

Much of work in Natural Language Processing is to decode the syntactic relationshp between words, phrases, sentences and documents as well as to understand the semantic relationshp between them. In this document, we focus on the syntax behind the languages.

1.1 Constituency

Syntactic constituency is the idea that groups of words can behave as single units, or *constituents.* Part of developing a grammar involves building an inventory of the constituents in the language. The constituents can be placed in one place together but not individual word within it. The basic constituents in English are phrases below:

- **Noun Phrase (NP)**: a sequence of words surrounding **at least one noun**. e.g. "Harry Potter", "they", "three parties from Brooklyn"; These groups of words can appear at a similar syntactic environments, for example, before a *verb.* While the group of words can all appear before a verb, this is not true of each of the individual words that make up a noun phrase. e.g. "*from* arrive". The entire group can also be reordered for preposed or postposed constructions in order to highlight the information.

- **preposed** or **postposed** constructions: Both are information structuring strategy, i.e. ways to **highlight** the information. **preposed**: moving elements to an earlier position in the clause. **postposed** moving elements to a later position in the clause.

- **Verb Phrase (VP)**: a syntactic unit composed of a **verb** and its **arguments** except the subject of an independent clause or coordinate clause. A verb phrase is similar to what is considered a **predicate** in traditional grammars. Verb phrases generally are divided among *two* types: **finite**, of which the head of the phrase is a **finite verb**; and **nonfinite**, where the head is a **nonfinite verb**, such as an *infinitive, participle or gerund.* Phrase *structure grammars* acknowledge both types, but *dependency grammars* treat the subject as just another verbal dependent, and they do not recognize the finite verbal phrase constituent. Understanding verb phrase analysis depends on knowing which theory applies in context.

- For example, "*prefer a morning flight*" is a VB, i.e. a verb followed by a NP. "*leave Boston in the morning*" is a VB, i.e. a verb followed by a NP and a prepositional phrase.

- **Prepositional Phrase (PP)**: A prepositional phrase generally has a preposition followed by a noun phrase. For example "*from Los Angeles*" is a PP, defined by preposition followed by a noun phrase.

- **Adjective Phrase (AP)**:

- **Determiner**: also called *determinative* (abbreviated *det*), is a word, phrase, or affix that occurs together with a *noun* or noun phrase and generally serves to **express the reference** of that noun or noun phrase in the context. That is, a determiner may indicate whether the noun is referring to a *definite* or *indefinite* element of a class, to a closer or more distant element, to

an element belonging to a specified person or thing, to a particular number or quantity, etc. Common kinds of determiners include **definite and indefinite articles** (like **the** and **a** or **an**), **demonstratives** (**this** and **that**), **possessive determiners** (**my** and **their**), **cardinal numerals**, **quantifiers** (many, both, all and no), **distributive determiners** (**each**, **any**), and **interrogative determiners** (which).

- **Nominal**: In linguistics, the term nominal refers to a **category** used to group together *nouns* and *adjectives* based on ***shared properties***. The motivation for nominal grouping is that in many languages nouns and adjectives share a number of morphological and syntactic properties.
- **Gerundive postmodifiers** are so called because they consist of a verb phrase that begins with the gerundive (-ing) form of the verb. "any of those [*leaving on Thursday*]"

1.2 Context-Free Grammars

The most widely used formal system for modeling ***constituent structure*** in English and other natural languages is the ***Context-Free Grammar***, or **CFG**. GFG are also called **Phrase-Structure Grammars**, and the formalism is equivalent to **Backus-Naur Form**, or **BNF**. A context-free grammar consists of a set of **rules** or **productions**, each of which expresses the ways that symbols of the language can be *grouped* and *ordered* together, and a **lexicon** of words and symbols. They are represented as

non-terminal symbol $\rightarrow$ an *ordered* list of **terminal** or **non-terminal** symbols.

For example, NP (or noun phrase) can be composed of either a *ProperNoun* or a *determiner* (Det) followed by a *Nominal*; a Nominal in turn can consist of one or more Nouns.

$$\begin{aligned} \text{NP} &\rightarrow \text{Det Nominal} \\ \text{NP} &\rightarrow \text{ProperNoun} \\ \text{Nominal} &\rightarrow \text{Noun}|\text{Nominal Noun} \\ \text{Det} &\rightarrow \text{a} \\ \text{Noun} &\rightarrow \text{flight} \end{aligned}$$

The symbols that are used in a CFG are divided into two classes.

- **terminal symbols**: The symbols that correspond to **words** in the language; the *lexicon* is the set of rules that introduce these terminal symbols.
- **non-terminal symbols**: The symbols that express *abstractions* over these terminals. The non-terminal associated with each word in the lexicon is its *lexical category*, or **part of speech**.

In each context-free rule, the item to the right of the arrow ($\rightarrow$) is an ordered list of one or more terminals and non-terminals; to the left of the arrow is a **single** non-terminal symbol expressing some cluster or generalization.

A CFG can be thought of in two ways: as a device for *generating sentences* and as a device for *assigning a structure* to a given sentence. As a generatior, we can view " $\rightarrow$ " as "rewrite the

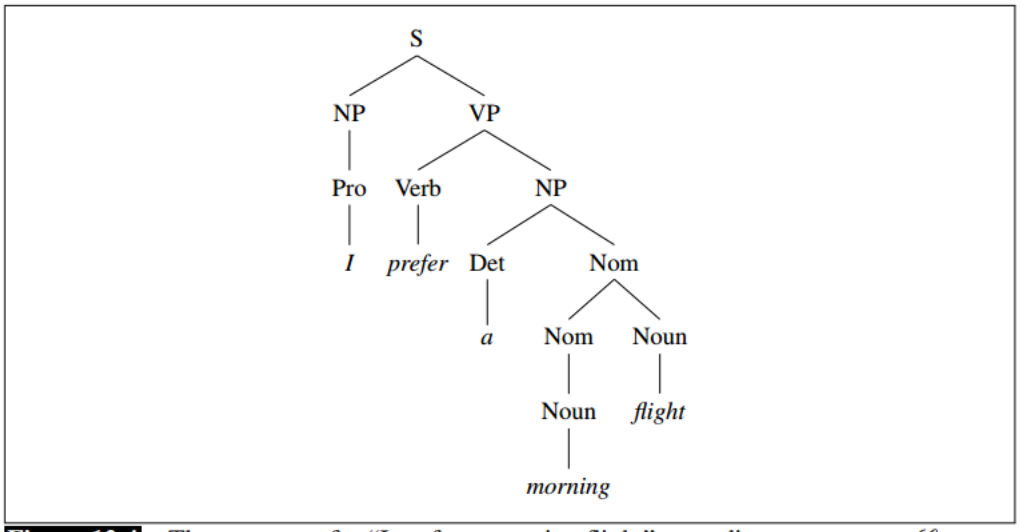

Figure 12.4 The parse tree for "I prefer a morning flight" according to grammar $\mathscr{L}_0$.

Figure 1: An example parse tree.

symbol on the left with the string of symbols on the right". This sequence of rule expansions is called a ***derivation*** of the string of words. CFG can be represented in two ways:

- ***parse tree***. Because context-free rules can be hierarchically embedded, we can combine the previous rules with others to build a tree-structure directed graph. The *formal* language defined by a CFG is the set of strings that are derivable from the designated **start symbol** (root). Each ***grammar*** must have *one* designated start symbol, which is often called *S*. Since context-free grammars are often used to define sentences, S is usually interpreted as the sentence node, and the set of strings that are derivable from S is the set of sentences in some simplified version of English. Figure 2 describes a grammar $\mathcal{L}_0$. We can use this grammar to *generate sentences* of this ATIS-language. We start with S, expand it to NP VP, then choose a random expansion of NP (lets say, to I), and a random expansion of VP (lets say, to Verb NP), and so on until we generate the string *I prefer a morning flight*.
- ***bracketed notation*** here is the bracketed representation of the parse tree of Figure 2:

$$[_S[_{NP}[_{Pro}\mathbf{I}]][_{VP}[_V\mathbf{prefer}][_{NP}[_{Det}\mathbf{a}][_{Nom}[_N\mathbf{morning}][_{Nom}[_N\mathbf{flight}]]]]]]$$

A CFG like that of above defines a *formal language*. A formal language is a set of strings. Sentences (strings of words) that can be derived by a grammar are in the formal language defined by that grammar, and are called ***grammatical*** sentences. Sentences that cannot be derived by a given formal grammar are not in the language defined by that grammar and are referred to as **ungrammatical**. In linguistics, the use of formal languages to model natural languages is called generative grammar since the language is defined by the set of possible sentences generated by the grammar.

As seen above, CFGs are set of rules to define the constituent structure of natural languages, i.e. how the words are grouped and ordered to form a sentence. These rules can be used to define the domain of distribution of a given language. CFG is a language model to generate formal languages. However, it does not care the meaning of the sentence. *A language is defined through the concept of derivation*. One string derives another one if it can be rewritten as the second one by some series

of rule applications.

- **syntactic parsing**: The problem of mapping from a string of words to its parse tree.

2 Grammar rules in English and some terminologies

2.1 English Sentence-Level Constructions

There are several common sentence-level constructions:

- **declarative structure**: Sentences with declarative structure have a subject noun phrase followed by a verb phrase. "S $\rightarrow$ NP VP". It is used to make *statements* or *assertions*. e.g. "You are my friend."
- **imperative structure**: Sentences with imperative structure often begin with a verb phrase and have no subject. "S $\rightarrow$ VP." They are called imperative because they are almost always used for *commands* and *suggestions*. e.g. "Be my friend!"
- **interrogative structure**, i.e. yes-no question structure: Sentence typically raises a *question*. Sentences with interrogative structure are often (though not always) used to ask questions; they begin with an auxiliary verb, followed by a subject NP, followed by a VP. "S $\rightarrow$ Aux NP VP." e.g. "Are you my friend?"
- **exclamative structure**, sometimes called an exclamatory sentence, typically expresses an exclamation;
- **wh-question structure**: These are so named because one of their constituents is a *wh-phrase*, that is, one that includes a *wh-word* (who, whose, when, where, what, which, how, why). It includes *wh-subject-question structure* and *wh-non-subject-question structure*.
  - **wh-subject-question structure** is identical to the *declarative* structure, except that the first noun phrase contains some wh-word. "S $\rightarrow$ Wh-NP VP". e.g. "What airlines fly from Burbank to Denver?"
  - **wh-non-subject-question structure**, the wh-phrase is not the subject of the sentence, and so the sentence includes another subject. In these types of sentences the auxiliary appears before the subject NP, just as in the *interrogative* structures. "S $\rightarrow$ Wh-NP Aux NP VP." e.g. "What flights do you have from Burbank to Tacoma Washington?" This type of structure contains what are called **long-distance dependencies** because the Wh-NP is far away from the predicate that it is semantically related to, the main verb have in the VP.

2.2 The Noun Phrase

An English noun phrase can have:

- **determiner**: Common kinds of determiners include **definite and indefinite articles** (like **the** and **a** or **an**), **demonstratives** (**this** and **that**), **possessive determiners** (**my** and **their**)

- **nominal**: The nominal construction follows the determiner and contains any pre- and post-head noun modifiers.
- **before the head noun**: A number of different kinds of word classes can appear *before* the head but *after the determiner* (the **postdeterminers**) in a nominal. These include **cardinal numbers**, **ordinal numbers**, **quantifiers**, and **adjectives**. Adjectives can also be grouped into a phrase called an ***adjective phrase*** or AP. APs can have an adverb before the adjective.
- **after the head noun**: A head noun can be followed by *postmodifiers* such as *prepositional phrases*, *non-finite clauses*, *relative clauses*. The three most common kinds of non-finite postmodifiers are the **gerundive (-ing), -ed, and infinitive forms**.
  - **Gerundive postmodifiers**: i.e. -ing form; e.g. "any flights [*arriving after eleven a.m.*]"
  - **non-finite clauses**: e.g. "She left the building [*to find her friends*]"
  - **infinitives forms**: e.g. "I need to have dinner [*served*]"
  - **-ed forms**: e.g. "Which is the aircraft [*used by this flight*]?"
  - A *postnominal* ***relative clause*** (more correctly a restrictive relative clause), is a clause that often begins with a relative pronoun (that and who are the most common). The relative pronoun functions as the subject of the embedded verb. e.g. "a flight [*that serves breakfast*]"
- **before the noun phrase**: Word classes that modify and appear before NPs are called **predeterminers**. Many of these have to do with number or amount; a common predeterminer is "all".

2.3 The Verb Phrase

The verb phrase consists of the verb and a number of other constituents.

$$\begin{aligned}
&\text{VP} \rightarrow \text{Verb} \\
&\text{VP} \rightarrow \text{Verb NP} \\
&\text{VP} \rightarrow \text{Verb PP} \\
&\text{VP} \rightarrow \text{Verb NP PP} \\
&\text{VP} \rightarrow \text{Verb S}
\end{aligned}$$

Many other kinds of constituents, such as an entire embedded sentence, can follow the verb. These are called ***sentential complements***, "VP $\rightarrow$ Verb S". Similarly, another potential constituent of the VP is another VP. This is often the case for verbs like "want", "would like", "try", "intend", "need": "Hi, I want [$_{\text{VP}}$ *to arrange three flights*]"

While a verb phrase can have many possible kinds of constituents, not every verb is compatible with every verb phrase. This idea that verbs are compatible with different kinds of complements is a very old one; traditional grammar distinguishes between ***transitive verbs*** like "find", which take a direct object NP ("I found a flight"), and ***intransitive verbs*** like "disappear", which do not ("I disappeared a flight"). We say that a verb like "find" **subcategorizes for** an NP, and a verb like "want" *subcategorizes for* either an NP or a non-finite VP. We also call these constituents the **complements of the verb** (hence our use of the term *sentential complement* above). So we

Grammar	Lexicon
$S \rightarrow$ *NP VP .*	*PRP* $\rightarrow$ *we* | *he*
$S \rightarrow$ *NP VP*	*DT* $\rightarrow$ *the* | *that* | *those*
$S \rightarrow$ *" S " , NP VP .*	*JJ* $\rightarrow$ *cold* | *empty* | *full*
$S \rightarrow$ *-NONE-*	*NN* $\rightarrow$ *sky* | *fire* | *light* | *flight* | *tomorrow*
NP $\rightarrow$ *DT NN*	*NNS* $\rightarrow$ *assets*
NP $\rightarrow$ *DT NNS*	*CC* $\rightarrow$ *and*
NP $\rightarrow$ *NN CC NN*	*IN* $\rightarrow$ *of* | *at* | *until* | *on*
NP $\rightarrow$ *CD RB*	*CD* $\rightarrow$ *eleven*
NP $\rightarrow$ *DT JJ , JJ NN*	*RB* $\rightarrow$ *a.m.*
NP $\rightarrow$ *PRP*	*VB* $\rightarrow$ *arrive* | *have* | *wait*
NP $\rightarrow$ *-NONE-*	*VBD* $\rightarrow$ *was* | *said*
VP $\rightarrow$ *MD VP*	*VBP* $\rightarrow$ *have*
VP $\rightarrow$ *VBD ADJP*	*VBN* $\rightarrow$ *collected*
VP $\rightarrow$ *VBD S*	*MD* $\rightarrow$ *should* | *would*
VP $\rightarrow$ *VBN PP*	*TO* $\rightarrow$ *to*
VP $\rightarrow$ *VB S*	
VP $\rightarrow$ *VB SBAR*	
VP $\rightarrow$ *VBP VP*	
VP $\rightarrow$ *VBN PP*	
VP $\rightarrow$ *TO VP*	
SBAR $\rightarrow$ *IN S*	
ADJP $\rightarrow$ *JJ PP*	
PP $\rightarrow$ *IN NP*	

Figure 12.10 A sample of the CFG grammar rules and lexical entries that would be extracted from the three treebank sentences in Fig. 12.7 and Fig. 12.9.

Figure 2: An example CFG rules from treebank.

say that "want" can take a **VP complement**. These possible sets of complements are called the **subcategorization frame** for the verb.

We can capture the association between verbs and their complements by making separate subtypes of the class Verb. Each VP rule could then be modified to require the appropriate *verb subtype*.

VP → Verb-with-no-complement
VP → Verb-with-NP-comp NP
VP → Verb-with-S-comp S

2.4 Coordination

The major phrase types discussed here can be conjoined with **conjunctions** like "and", "or", and "but" to form larger constructions of the same type. For example, a **coordinate** noun phrase can consist of two other noun phrases separated by a conjunction. e.g. "A and B".

Note that the ability to *form coordinate phrases through conjunctions* is often used as a **test for constituency**. The fact that these phrases can be conjoined is evidence for the presence of the

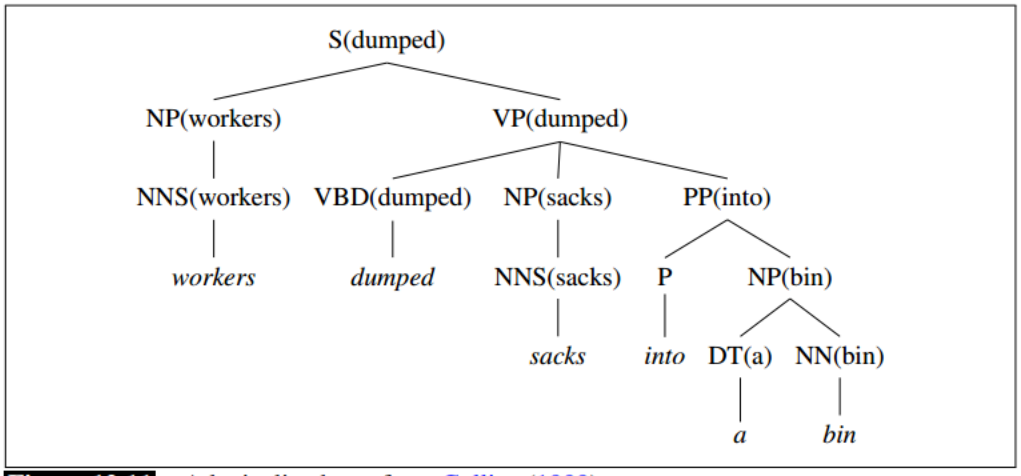

Figure 12.11 A lexicalized tree from Collins (1999).

Figure 3: An example head passing up to the tree.

underlying *Nominal constituent* we have been making use of.

I need to know the $[_{Nom}[_{Nom}$ aircraft] and$[_{Nom}$ flight number]].

The rules for these constituents can be summarized as below:

$$\text{VP} \rightarrow \text{VP and/or/but VP}$$
$$\text{S} \rightarrow \text{S and/or/but S}$$
$$\text{NP} \rightarrow \text{NP and/or/but NP}$$
$$\text{Normial} \rightarrow \text{Normial and/or/but Normial}$$

3 Treebanks and Head-finding rules

The syntactic constituents could be associated with a **lexical head**; N is the head of an NP, V is the head of a VP. This idea of a head for each constituent dates back to Bloomfield 1914, and is central to the **dependency grammars** and **dependency parsing**.

In one simple model of lexical heads, each context-free rule is associated with a *head*. The head is the word in the phrase that is grammatically the most *important*. **Heads are passed up the parse tree**; thus, each non-terminal in a parse tree is annotated with a single word, which is its *lexical head*. Figure 3 shows an example of such a tree from Collins (1999), in which each non-terminal is annotated with its head.

For the generation of such a tree, each CFG rule must be augmented to identify *one right-side constituent* to be the **head child**. The *headword* for a node is then set to the headword of its head child.

An alternative approach to finding a head is used in most practical computational systems. Instead of specifying head rules in the grammar itself, heads are identified *dynamically* in the context of trees for specific sentences. In other words, once a sentence is parsed, the resulting tree is walked to **decorate** each node with the appropriate head.

Parent	Direction	Priority List
ADJP	Left	NNS QP NN $ ADVP JJ VBN VBG ADJP JJR NP JJS DT FW RBR RBS SBAR RB
ADVP	Right	RB RBR RBS FW ADVP TO CD JJR JJ IN NP JJS NN
PRN	Left	
PRT	Right	RP
QP	Left	$ IN NNS NN JJ RB DT CD NCD QP JJR JJS
S	Left	TO IN VP S SBAR ADJP UCP NP
SBAR	Left	WHNP WHPP WHADVP WHADJP IN DT S SQ SINV SBAR FRAG
VP	Left	TO VBD VBN MD VBZ VB VBG VBP VP ADJP NN NNS NP

Figure 12.12 Some head rules from Collins (1999). The head rules are also called a **head percolation table**.

Figure 4: The rule for head-finding in a constituency tree.

Selected other rules from this set are shown in Figure 4. For example, for VP rules of the form $\text{VP} \rightarrow \text{Y}_1 \ldots \text{Y}_n$, the algorithm would start from the left of $\text{Y}_1 \ldots \text{Y}_n$ looking for the first Y_i of type TO; if no TOs are found, it would search for the first Y_i of type VBD; if no VBDs are found, it would search for a VBN, and so on.

4 Grammar Equivalence and Normal Form

We can compare if two grammars are the same. We usually distinguish two kinds of grammar equivalence:

- **strong equivalence**: Two grammars are strongly equivalent if they ***generate*** *the same set of strings* and if they ***assign*** *the same phrase structure* to each sentence (allowing merely for renaming of the non-terminal symbols).
- **weak equivalence**: Two grammars are weakly equivalent if they ***generate*** *the same set of strings* but do **not** assign the same phrase structure to each sentence.

It is sometimes useful to have a **normal form** for grammars, in which each of the productions takes a particular form.

Chomsky normal form (CNF): for a context-free grammar to be CNF, if it is ϵ-free and if in addition each production is either of the form $A \rightarrow B\ C$ or $A \rightarrow a$. That is, the right-hand side of each rule either has **two non-terminal symbols** or **one terminal symbol**. CNF implies that the parsing tree is **binary tree** since it forms **binary branching**.

[**Theorem**]: *Any context-free grammar can be converted into a weakly equivalent Chomsky normal form grammar.*

The generation of a symbol A with a potentially infinite sequence of symbols B with a rule of the form $A \rightarrow A\ B$ is known as **Chomsky-adjunction**.

References

Dan Jurafsky and James H Martin. Speech and language processing. vol. 3. *US: Prentice Hall*, 2014.