CSE 396 Lecture Thu Mar 8 Spring 2018 (1)

Top Hat #4268

A context-free grammar (CFG) – an object – is a 4-tuple $G = (V, \Sigma, R, S)$ where

- V is a finite alphabet of <u>variables</u>, aka <u>nonterminals</u> non terminals
- Σ, <u>also called T</u>, is a finite alphabet of <u>terminals</u>

We suppose $V \cap \Sigma = \emptyset$

- S, a member of V, is the <u>starting variable</u> aka. the <u>start</u> <u>symbol</u>
- R is a finite set of <u>rules</u> of the form

$$A \to X \quad \text{where } A \in V \text{ and } X \in (\Sigma \cup V)^*.$$

Really $R \subseteq V \times (V \cup \Sigma)^*$. Also called P for <u>production</u>

<u>Defn</u>: Given $X, Y \in (\Sigma \cup V)^*$, we write

$$X \underset{G}{\Longrightarrow} Y$$ "X <u>derives</u> Y in one step of G"

if X can be broken as $X = U \cdot A \cdot W$ and there is a rule $A \to Z$ in R s.t. $Y = U \cdot Z \cdot W$.

Also define $X \underset{G}{\Rightarrow}^{0} X$ for any $X \in (\Sigma \cup V)^*$ and for $K \geq 1$, $X \underset{G}{\Rightarrow}^{K} Z$ if there is a Y in $(\Sigma \cup V)^*$ st. $X \underset{G}{\Rightarrow}^{K-1} Y$ and $Y \underset{G}{\Rightarrow} Z$.

I.e., "X derives Z in K steps of the grammar G."

<u>Finally:</u> $\boxed{X \underset{G}{\Rightarrow}^{*} Z \text{ if } X \underset{G}{\Rightarrow}^{K} Z \text{ for some } K \geq 0.}$

Language has Terminal Strings only

$\boxed{L(G) = \{w \in \Sigma^* : S \underset{G}{\Rightarrow}^{*} w\}.}$

However: A string $X \in (\Sigma \cup V^*)$ is called a <u>Sentential</u> <u>Form</u> of G if $S \underset{G}{\Rightarrow}^{*} X$.

Origin Noam Chomsky, 1956-57 Book: <u>Syntactic Structures</u>:

"followed by a"

$S \rightarrow \underbrace{N}_{\text{Noun Phrase}}\ \underbrace{V}_{\text{Verb Phrase}}$

A sentence can be a Noun Phrase Verb Phrase

Variables are often written as "tokens" <noun_phrase> as in <verb_phrase> <u>BNF</u>

Examples: $L = \underline{\underline{PAL}} = \{x \in \{0,1\}^* : x = \underline{\underline{x^R}}\}$

$\varepsilon^R = \varepsilon$ x reversed: $01^R = 10$ $0110^R = 0110$

Compare to final grammar in Tue 3/7 lecture

"or" for rule options.

$G = S \to \varepsilon \mid 0S0 \mid 1S1$

Understood: $V = \{S\}$, $\Sigma = \{0,1\}$, rules are $S \to \varepsilon$, $S \to 0S0$, $S \to 1S1$

Is $L(G) \subseteq L$? "Is G sound for L?"

Does $L(G) \supseteq L$? "Is G comprehensive?"

Is G correct? $L(G) = L$.

Is this G sound? Let's try some derivations:

$S \underset{G}{\Rightarrow} \varepsilon$. $\varepsilon \in \Sigma^*$. Is $\varepsilon \in L$? yes ✓

$S \underset{G}{\Rightarrow} 0S0 \underset{G}{\Rightarrow} 0 \cdot \varepsilon \cdot 0 = \underline{00} \in \Sigma^*$ Is $00 \in PAL$? yes ✓

Similarly, $S \underset{G}{\Rightarrow} 1S1 \underset{G}{\Rightarrow} 11$ $11 \in PAL$? yes.

$S \Rightarrow 0S0 \Rightarrow 0 \cdot 1S1 \cdot 0 \Rightarrow 0 \cdot 11 \cdot 0 = 0110$. yes

Is G comprehensive for PAL? **No**: G cannot derive the palindromes $010, 101, 000, 111$, nor 0. or 1 for that matter.

The crispest way to show this is to see that G obeys a stronger soundness condition: every $x \in \Sigma^*$ it derives is an even-length palindrome.

Fundamental Example: Balanced Parentheses

$\Sigma = \{ \text{'('}, \text{')'} \}$. Consider two grammars:

$G_1: S_1 \to (S_1) \mid S_1 S_1 \mid \varepsilon$

( ) ? as basis?

EXTRA STUDY Q: If we change the ε to "( )" then are the modified CFGs comprehensive for nonempty balanced-paren strings?

$G_2: S_2 \to (S_2) S_2 \mid \varepsilon$

Soundness of G_1 does imply soundness of G_2 because G_1 can simulate the rule $S_2 \to (S_2) S_2$ via its derivation $S_1 \Rightarrow S_1 S_1 \Rightarrow (S_1) S_1$.

Hence G_2 is also sound. ¿ Are G_1 and G_2 comprehensive? I.e. is every balanced string of parens derivable in G_2, and hence derivable in G_1? $\boxed{\text{YES}}$

IMHO this is easier for G_2 with the parsing intuition that the lead char (of a nonempty x) must be a '(' and the ')' in the rule $S_2 \to (S_2) S_2$ will be chosen as its "matching mate" and so a

This is ingrained in the example of (Fully-Parenthesized) <u>Formulas</u>:

text: $\underline{a}$.

$$E \longrightarrow \langle\text{constant}\rangle \mid \langle\text{variable}\rangle \mid (E)$$

↑ start symbol, "Expression"

can be a

numeric constants with digits and . as terminals, and identifier letters as variables not grammar variables.

allows extra parens.

$$E \rightarrow (E+E) \mid (E-E) \mid (E*E) \mid (E/E)$$

The interpretation of E as "I am a legal expression" shows this grammar is sound.

It does not, however, cover all expressions we tend to write using <u>precedence</u> in place of parentheses: $(X + Y*Z)$

However, CFGs <u>can</u> convey precedence as well as <u>balance</u>. → next lecture, "ETF" example in text (expression term factor) ~~expression term formula~~