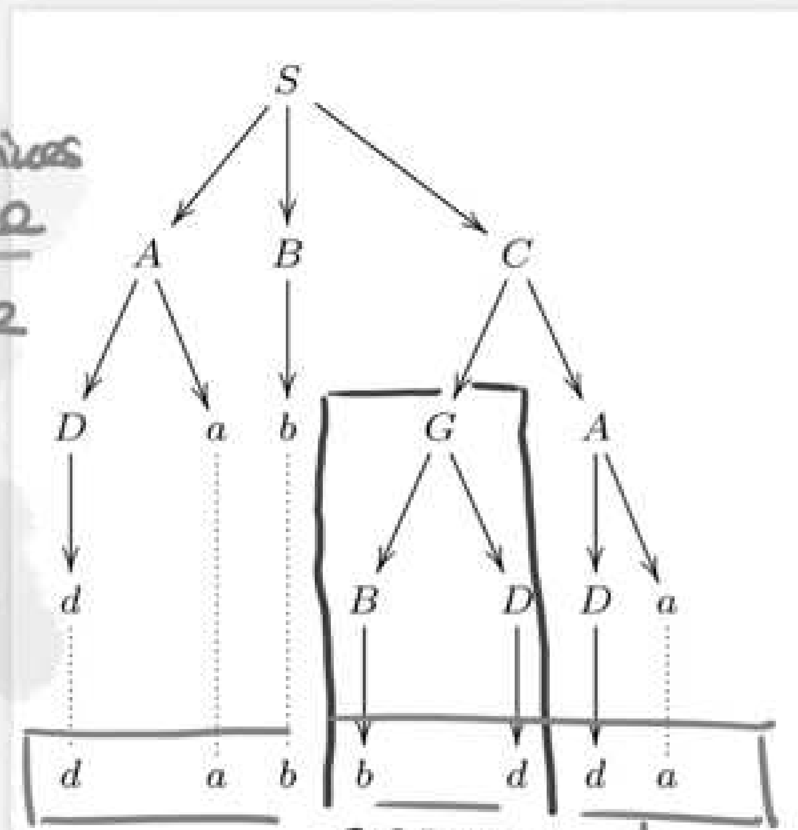
XII TWELFTH LECTURE
AUTOMATA & FORMAL LANGUAGES

1 November 2023

Lecture $\overline{XI}$: TREES & LABELLED TREES

G-PARSE TREES



SUBTREE

NOTE : A G-derivation determines a unique G-parse tree

BUT a G-parse tree does not determine a unique G-derivation (only up to reordering)

Parse tree $\mathbb{T}$ determines string $\sigma_{\mathbb{T}}$, read off at the leaves in left-to-right order.

$\sigma_{\mathbb{T}} =$ dabbdda.

If $t \in \mathbb{T}$, $\sigma_{\mathbb{T}} = \alpha \sigma_{\mathbb{T}_t} \beta$, some α, β.

GRAFTING If $\ell(t) = A$ & $\mathbb{T}'$ parse tree from A :

$$\sigma_{\mathbb{T}} = \alpha \sigma_{\mathbb{T}_t} \beta \longrightarrow \sigma_{\mathrm{graft}(\mathbb{T}, t, \mathbb{T}')} = \alpha \sigma_{\mathbb{T}'} \beta.$$

So, in particular, if $\sigma_{\mathbb{T}} = xyz \in \mathcal{L}(G)$ $\quad y = \sigma_{\mathbb{T}_t}$

$\sigma_{\mathbb{T}'} = v$. Then $xvz \in \mathcal{L}(G)$.

CHOMSKY NORMAL FORM CNF

Remember that in general, there is no bound on the length of a derivation for w that only depends on $|w|$.

Noam Chomsky

Chomsky in 2017

Born	Avram Noam Chomsky December 7, 1928 (age 94) Philadelphia, Pennsylvania, U.S.
Spouses	Carol Schatz (m. 1949; died 2008) Valeria Wasserman (m. 2014)
Children	3, including Aviva
Parent	William Chomsky (father)

Definition We say G is in CNF if all production rules are of the form

$$A \longrightarrow a \quad \text{terminal}$$

or

$$A \longrightarrow BC \quad A, B, C \in V \quad \text{nonterminal.}$$

If G is in CNF, it is context-free.

The G-parse trees for a CNF grammar are very special:

every node is

(a) binary branching with non-leaf succ. or

(b) single successor which is a leaf

(c) leaf.

Lemma If G is in CNF and $w \in \mathcal{L}(G)$, then every G-derivation of w has length $2|w|-1$.

Proof. Note there are only two types of rules:

terminal $(A \to a)$: preserves length of string
increases number of letters by 1
decreases # of variables by 1.

non-terminal $(A \to BC)$: increases length by 1
preserves # of letters.

Clearly if $w \in \mathcal{L}(G)$, it must use exactly $|w|$ many terminal rules.
Since $|S|=1$, we need $|w|-1$ nonterminal rules to achieve the right length.

So total length is $|w| + |w| - 1 = 2|w|-1$.

q.e.d.

<u>Lemma</u> If G CNF and $\mathbb{T}$ is a G-parse tree of w of height $h+1$.
Then $|w| \leq 2^h$.

<u>Proof.</u> Note that T is binary branching and a binary branching tree of height $h+1$ can have at most 2^{h+1} leaves.

So clearly $|w| \leq 2^{h+1}$.

But in a G-parse tree, branching nodes cannot have leaves as successors, & so there are at most 2^h leaves.

q.e.d.

THEOREM (Chomsky).
If G is context-free, then there is an equivalent G' in CNF.

Definition

A rule $A \longrightarrow \alpha$ is called <u>variable-targeted</u> if all symbols on RHS are variables.

A rule $A \longrightarrow B$ is called <u>unit production</u>

G is called <u>variable-targeted</u> if every rule is either v-t or terminal of the form $A \longrightarrow a$

G is called <u>unit-closed</u> if whenever

$$\left.\begin{array}{l} A \longrightarrow B \in P \\ B \longrightarrow \alpha \in P \end{array}\right\} \text{ then } A \longrightarrow \alpha \in P.$$

Lemma (L3.6) For each c-f grammar G there is an equivalent variable-targeted G'.

Proof. For each $a \in \Sigma$ add new variable X_a. For α, write $X(\alpha)$ where all letters a are replaced with X_a.

$$P' := \{A \longrightarrow X(\alpha) ;\ A \longrightarrow \alpha \in P\} \cup \{X_a \longrightarrow a ;\ a \in \Sigma\}$$

Then G' is eq. to G. q.e.d.

__Lemma__ (L 3.7)

For each c-f grammar G there is an equivalent grammar that is unit closed.

__Proof.__ If $A \longrightarrow B$, $B \longrightarrow \alpha \in P$, we can add $A \longrightarrow \alpha$ to the grammar w/o changing language [ute G is c-f].

Note that it may not be enough to add all of these instances to get unit closure since you may create new instances:

$$\left.\begin{array}{l} A \longrightarrow B \\ \left.\begin{array}{l} B \longrightarrow C \\ C \longrightarrow \alpha \end{array}\right\} B \longrightarrow \alpha \end{array}\right] A \to \alpha$$

Q: Can this go on forever?

A: No, since everything that's added is of the form

$A \longrightarrow \alpha$ where $A \in V$ & α is a RHS of some rule in P.

Thus we add at most $|V||P|$ many new rules before we reach unit closure.

q.e.d.

Lemma (L 3.8)

If G is unit closed and G' is G with a unit productions removed, then G & G' are equivalent.

Proof. Clearly, $\mathcal{L}(G') \subseteq \mathcal{L}(G)$.

For $\supseteq$, we show that the shortest G-derivation of a word $w \in \mathcal{L}(G)$ does not use unit productions.

$$S \xrightarrow[n]{G} \underbrace{\boxed{\alpha A \beta \xrightarrow[1]{G} \alpha B \beta}}_{\text{UNIT PRODUCTION}} \xrightarrow[k+l]{G} \gamma B \delta \xrightarrow[1]{G} \gamma \xi \delta \xrightarrow[m]{G} w$$

$\alpha \xrightarrow{G} \gamma$ k steps

$\beta \xrightarrow{G} \delta$ l steps

$k+l$

$n+m+k+l+2$

$A \to B \in P$

$B \to \xi \in P$

By unit closure $A \to \xi \in P$.

$$\underbrace{S \xrightarrow[n]{G} \alpha A \beta \xrightarrow[1]{G} \alpha \xi \beta \xrightarrow[k]{G} \gamma \xi \beta \xrightarrow[l]{G} \gamma \xi \delta \xrightarrow[m]{G} w}_{n+m+k+l+1 \,<\, n+m+k+l+2.}$$

So the first derivation was not minimal. q.e.d.

Lemma 3.9 If G is a c-f grammar and $A \longrightarrow B_0 \ldots B_n$ is variable-targeted, then add new variable X_i and write rules

$$A \longrightarrow B_0 X_0$$
$$X_0 \longrightarrow B_1 X_1$$
$$\vdots$$
$$X_{n-1} \longrightarrow B_{n-1} B_n$$

Then the grammar G' with $A \longrightarrow B_0 \ldots B_n$ replaced with these n rules is eq. to G.

Proof. Obviously, $\mathcal{L}(G') \supseteq \mathcal{L}(G)$.
For the other direction observe that every G'-derivation either uses no new rules (i.e., it is a G-derivation) or it uses all n rules in precisely that order.

Compare (S) on ES#1.

q.e.d.

PROOF OF CHOMSKY'S THEOREM

Take c-f grammar $G = G_0$:

STEP 1 Use L 3.6 to make it variable-targeted. $\mathcal{L}(G_0) = \mathcal{L}(G_1)$.
G_1

Now all rules are $A \longrightarrow a$, $A \longrightarrow \alpha$
$\alpha \in V^*$

STEP 2 Use L 3.7 to form unit closure G_2
$\mathcal{L}(G_1) = \mathcal{L}(G_2)$

STEP 3 Remove unit productions from G_2 and obtain G_3.
Lemma 3.8 : $\mathcal{L}(G_3) = \mathcal{L}(G_2)$.

STEP 4 Systematically replace all rules $A \longrightarrow \alpha$ with $|\alpha| > 2$ with $|\alpha|$ many non-term. CNF rules G_4

L 3.9: $\mathcal{L}(G_3) = \mathcal{L}(G_4)$.

Then G_4 is in CNF.

q.e.d.

(IMPORTANT) REMARK:
This is not just an existence proof, but an algorithm that produces G_4 from G_0.

THE PUMPING LEMMA FOR CONTEXT-FREE LANGUAGES

We already know that in general, the RPL cannot hold for c-f languages.

Lecture $\overline{\text{XI}}$, page 1.

CONTEXT-FREE RULE

$A \longrightarrow \beta$ where $\beta \neq \varepsilon$, $\beta \in \Omega^*$.

Example $L = \{0^n 1^n ; n > 0\}$

Grammar: $S \longrightarrow 0S1$, $S \longrightarrow 01$

Not regular due to (R)PL.

A.K.A. Bar-Hillel Lemma

Definition 3.10. Let $L \subseteq \mathbb{W}$ be a language. We say that L *satisfies the (context-free) pumping lemma with pumping number* n if for every word $w \in L$ such that $|w| \geq n$ there are words u, v, x, y, z such that $w = xuyvz$, $|uv| > 0$, $|uyv| \leq n$ and for all $k \in \mathbb{N}$, we have that $xu^k yv^k z \in L$. We say that L *satisfies the (context-free) pumping lemma* if there is some n such that it satisfies the (context-free) pumping lemma with pumping number n.

Instead of $w = xyz$ with pump y, we have

$x u y v z = w$

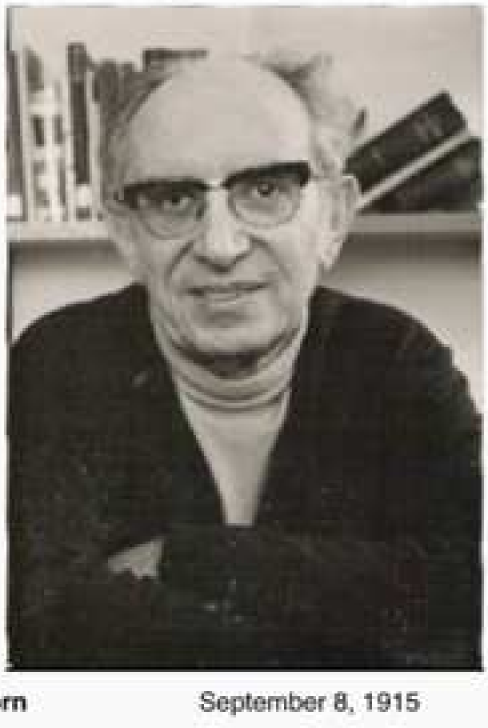
with pump in two parts: u & v

Result of pumping.

$x u^k y v^k z$

The location of the pump is not as fixed as in the RPL.

Yehoshua Bar-Hillel



Born	September 8, 1915 Vienna, Austria-Hungary
Died	September 25, 1975 (aged 60) Jerusalem, Israel