AUTOMATA & FORMAL LANGUAGES

Secoud Lectore

8 October 2022

RECAP

- If X is countable, then X^* and $\mathrm{Fin}(X)$ are countable.
- If X is infinite, then $\wp(X)$ is uncountable.
- Notation for strings.

§ 1.2 REWRITE SYSTEMS

Ω finite set of symbols

Ω^* Ω-strings

USUALLY $\alpha, \beta, \gamma, \delta, \sigma, \tau$

We call elements of $\Omega^* \times \Omega^*$

REWRITE RULES

(PRODUCTION RULES)

We write suggestively $\alpha \longrightarrow \beta$ for (α, β)

INFORMAL INTERPRET.:

If α occurs in a string, I can rewrite it as β.

SEARCH-AND-REPLACE

Def. A pair $R = (\Omega, P)$ is called a REWRITE SYSTEM if P is a finite set of rewrite rules.

Prop. 1.5 There are only countably many rewrite systems on a fixed finite Ω.

Proof. Every P is an element of

$$\mathrm{Fin}(\Omega^* \times \Omega^*),$$

so it follows directly from P 1.1 and P 1.3. q.e.d.

If $R = (\Omega, P)$ is a rewrite system, and $\sigma, \tau \in \Omega^*$, we write

$$\sigma \xrightarrow{R}_1 \tau$$

$$:\Longleftrightarrow$$

$\exists \alpha, \beta, \gamma, \delta \in \Omega^*$ s.t.

$\sigma = \alpha\gamma\beta \quad \tau = \alpha\delta\beta$

$\gamma \longrightarrow \delta \in P.$

PRONOUNCED AS
σ is rewritten to τ in one step
OR
R produces τ from σ in one step

Now $\xrightarrow{R}$ is defined as the reflexive and transitive closure of $\xrightarrow{R}_1$; i.e.,

$$\sigma \xrightarrow{R} \tau \quad :\iff$$

$\sigma = \tau$ OR

there is $\sigma_0, \dots, \sigma_n$ s.t.

$\sigma_0 = \sigma$ $\quad \sigma_n = \tau$ and

$$\underbrace{\sigma_0 \xrightarrow{R}_1 \sigma_1 \xrightarrow{R}_1 \cdots \xrightarrow{R}_1 \sigma_n}_{\text{CALLED A R-DERIVATION OF } \tau \text{ FROM } \sigma.}.$$

We say the derivation has length n. [NOTE: It has $n+1$ strings!]

Write $\mathcal{D}(R,\sigma) := \{\tau \in \Sigma^*;\ \sigma \xrightarrow{R} \tau\}$

The set of strings that can be produced/derived/rewritten from σ.

§ 1.3 What about actual languages?

In language:

Ω letters $\rightsquigarrow$ Ω^* words
$\{a, b, c, \ldots, z\}$

Ω words $\rightsquigarrow$ Ω^* sentences

Ω sentences $\rightsquigarrow$ Ω^* texts

Not every element of Ω^* is a word / sentence / novel. The task is to describe which ones are

WELLFORMED

Actual human languages are

FINITE.

Normally determined by a finite set

$$D \subseteq \Omega^*,$$

the dictionary.

Noam CHOMSKY
born 1928

One of the most important features of human language is

LINGUISTIC RECURSION.

[The] arbitrary decree that there is a finite upper limit to sentence length in English ... would serve no useful purpose. ... The point is that there are processes of sentence formation that this elementary model for language is intrinsically incapable of handling. ... In general, the assumption that languages are infinite is made for the purpose of simplifying the description. If a grammar has no recursive steps, ... it will be prohibitively complex. If it does have recursive devices, it will produce infinitely many sentences.[4]

While there is a limit to human comprehension capacity, it's not clear where it lies. If we understand a set with k nestings, then we'll understand one with k+1 nestings.

LINGUISTIC RECURSION.

Whenever we have a sentence of English X, we can write "E observes that" in front of it and obtain another grammatical sentence.

B likes *A*.
C believes that *B* likes *A*.
D reports that *C* believes that *B* likes *A*.
E observes that *D* reports that *C* believes that *B* likes *A*.
etc.

Therefore: do not describe languages by dictionaries, but by rewrite rules.

Famous Chomsky examples:

nonsensical but grammatical

COLOURLESS GREEN IDEAS SLEEP FURIOUSLY.

nonsensical and ungrammatical

FURIOUSLY SLEEP IDEAS GREEN COLOURLESS.

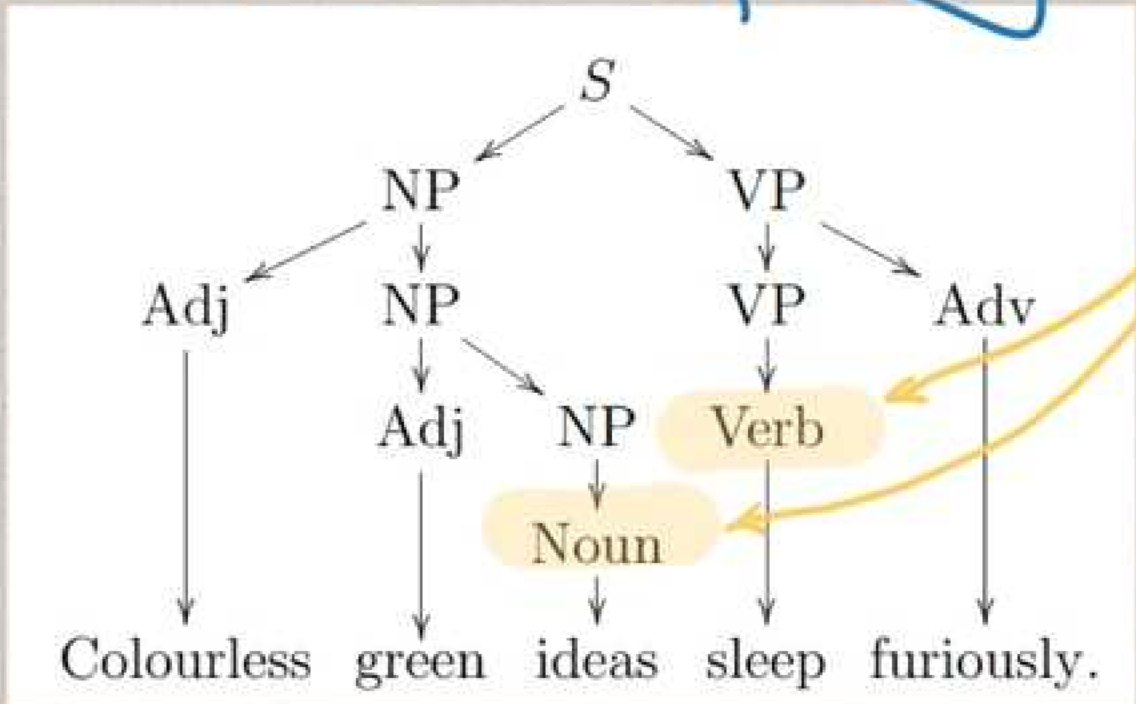
NOTE: The images on this page were corrected after the lecture. The images in the lecture were missing the yellow steps.

Example.

GENERATIVE GRAMMARS

$$
\begin{aligned}
S &\to \mathrm{NP}\,\mathrm{VP}, \\
\mathrm{NP} &\to \mathrm{Adj}\,\mathrm{NP}, \\
\mathrm{NP} &\to \mathrm{Noun}, \\
\mathrm{VP} &\to \mathrm{VP}\,\mathrm{Adv}, \\
\mathrm{VP} &\to \mathrm{Verb},
\end{aligned}
\tag{*}
$$

PARSE TREE (formally defined in Chapter 3).



On moodle: a proof that this grammar cannot derive the non-grammatical Chomsky sentence.

§1.4 Grammars

$$\Omega = \underbrace{\Sigma}_{\substack{\text{alphabet} \\ \text{elements: letters} \\ \text{terminal symbols}}} \cup \underbrace{V}_{\substack{\text{elements: variables} \\ \text{nonterminal} \\ \text{symbols}}}$$

with $\Sigma \cap V = \emptyset$
$\Sigma \neq \emptyset \neq V$

$a, b, c, \ldots$
small characters for letters

$A, B, C, \ldots$
capital characters for variables

$$\Sigma^* \subseteq \Omega^*$$

Elements of Σ^* are called WORDS.
We write $\mathbb{W}$ for Σ^*.
and $\mathbb{W}^+ := \Sigma^* \setminus \{\varepsilon\}$.

Any subset $L \subseteq \mathbb{W}$ is called a language.

Thus, by P 1.2. over any alphabet Σ, there are uncountably many languages.

Definition $G=(\Sigma, V, P, S)$ is called a grammar if Σ, V are nonempty disjoint sets with $\Omega := \Sigma \cup V$, and $(\Omega, P) =: R$ is a rewrite system and $S \in V$.

START SYMBOL

Since grammars are essentially rewrite systems, our notation carries over:

So:
$$\mathcal{D}(G,\sigma) := \mathcal{D}(R,\sigma)$$
$$\sigma \xrightarrow{G}_1 \tau \;:\Longleftrightarrow\; \sigma \xrightarrow{R}_1 \tau$$
$$\sigma \xrightarrow{G} \tau \;:\Longleftrightarrow\; \sigma \xrightarrow{R} \tau$$

We define
$$\mathcal{L}(G) := \mathcal{D}(G,S) \cap W.$$
called the language generated by G.

Warm-up.

(1) If there is no rule of the form

$$S \longrightarrow \alpha$$

in P, then

$$\mathcal{D}(G,S) = \{S\}$$

and thus $\mathcal{L}(G) = \emptyset$.

(2) If there is no rule of the form $\alpha \longrightarrow w$ for some $w \in W$, then $\mathcal{L}(G) = \emptyset$.

[Usually, words will ~~be~~ called u, v, w.]

Example

$\Sigma = \{a\}$

$V = \{S\}$

increases length by 2 | keeps length

$P_0 = \{S \longrightarrow aaS, S \longrightarrow a\}$

$G_0 = (\Sigma, V, P, S)$

Claim $\mathcal{L}(G_0) = \{a^{2n+1};\ n \in \mathbb{N}\}$

["$\subseteq$": Every elt of $\mathcal{D}(G, S)$ is of odd length

"$\supseteq$": Just produce a^{2n+1} by applying $S \longrightarrow aaS$ n times and $S \longrightarrow a$ one times.]

$$
\begin{aligned}
P_1 &:= \{S \to aSa, S \to a\},\\
P_2 &:= \{S \to Saa, S \to a\},\\
P_3 &:= \{S \to aaS, S \to aaSaa, S \to a\},\\
P_4 &:= \{S \to aaS, S \to Saa, S \to aSa, S \to a\},\\
P_5 &:= \{S \to aaS, aSa \to aaa, S \to a\},\\
P_6 &:= \{S \to aaS, aaS \to aSa, S \to a\}\\
P_7 &:= \{S \to aaS, aaS \to a, S \to a\}, \text{ etc.}
\end{aligned}
$$

By analysing the argument, we observe that

$G_1, G_2, G_3, \ldots, G_7$

with the appropriate P_i produce the same language. $G_i := (\Sigma, V, P_i, S)$

$\longrightarrow$ EQUIVALENCE OF GRAMMARS

$\boxed{G \,\&\, G' \text{ are } \underline{\text{equivalent}} \text{ if } \mathcal{L}(G) = \mathcal{L}(G').}$