Automata Theory :: Unrestricted Grammars

Jörg Endrullis

Vrije Universiteit Amsterdam

Unrestricted Grammars

What class of grammars corresponds to Turing machines?

An unrestricted grammar G contains rules

$$x \longrightarrow y$$

where $x \neq \lambda$.

Note that there is no restriction other than x being non-empty.

Theorem

A language L is generated by an unrestricted grammar $\iff$ L is accepted by a Turing machine.

From Unrestricted Grammars to Turing Machines

Theorem

For every unrestricted G there is a Turing machine M such that

$$L(M) = L(G)$$

Construction

Input for M is a word w (written on the tape).

M can do a **breadth-first search** for a derivation of w from S.

If a derivation is found, then w is accepted by M.

Then $L(M) = L(G)$.

From Turing Machines to Unrestricted Grammars

Theorem

For every TM M there is a grammar G with $L(G) = L(M)$.

Construction

The variables are S, T, $\square$
and V_{γ}^{α}, $V_{q\gamma}^{\alpha}$ for every $\alpha \in \Sigma \cup \{\square\}$, $\gamma \in \Gamma$ and $q \in Q$.

Step 1: guessing the word w

$$S \to V_{\square}^{\square} S \mid S V_{\square}^{\square} \mid T$$
$$T \to T V_{a}^{a} \mid V_{q_0 a}^{a} \qquad \text{for every } a \in \Sigma$$

After step 1, we have derived something of the form

$$V_{\square}^{\square} \cdots V_{\square}^{\square} V_{q_0 a_1}^{a_1} V_{a_2}^{a_2} V_{a_3}^{a_3} \cdots V_{a_n}^{a_n} V_{\square}^{\square} \cdots V_{\square}^{\square}$$

where $w = a_1 a_2 \cdots a_n$.

Next, the TM is simulated using the lower line (the subscripts).

From Turing Machines to Unrestricted Grammars

Construction continued

Step 2: simulating the TM (in the subscripts)

$$V^{\alpha}_{qc} V^{\beta}_{\gamma} \rightarrow V^{\alpha}_{d} V^{\beta}_{q'\gamma} \qquad \text{if } \delta(q, c) = (q', d, R)$$

$$V^{\beta}_{\gamma} V^{\alpha}_{qc} \rightarrow V^{\beta}_{q'\gamma} V^{\alpha}_{d} \qquad \text{if } \delta(q, c) = (q', d, L)$$

for every $\alpha, \beta \in \Sigma \cup \{\Box\}$ and $\gamma \in \Gamma$.

Step 3: If TM reaches accepting state, then generate w.
(From the superscripts left unchanged in step 2.)

$$V^{\alpha}_{q\gamma} \rightarrow \alpha \qquad \text{for every } q \in F$$

$$\beta V^{\alpha}_{\gamma} \rightarrow \beta\alpha$$

$$V^{\alpha}_{\gamma}\beta \rightarrow \alpha\beta$$

$$\Box \rightarrow \lambda$$

for every $\alpha, \beta \in \Sigma \cup \{\Box\}$ and $\gamma \in \Gamma$.

Then $L(G) = L(M)$.

Example

Consider the TM with $\Sigma = \{a, b, c\}$, $\Gamma = \Sigma \cup \{\Box\}$, $F = \{q_2\}$ and

$$\delta(q_0, a) = (q_0, c, R) \qquad \delta(q_0, c) = (q_1, b, L)$$
$$\delta(q_0, b) = (q_0, b, R) \qquad \delta(q_1, b) = (q_2, a, R)$$

This TM accepts the language $L((a+b)^* bc(a+b+c)^*)$.

The resulting grammar is:

$$S \to V_{\Box}^{\Box} S \mid S V_{\Box}^{\Box} \mid T$$
$$T \to T V_a^a \mid T V_b^b \mid T V_c^c \mid V_{q_0 a}^a \mid V_{q_0 b}^b \mid V_{q_0 c}^c$$

$$V_{q_0 a}^{\alpha} V_{\gamma}^{\beta} \to V_c^{\alpha} V_{q_0 \gamma}^{\beta} \qquad V_{q_2 \gamma}^{\alpha} \to \alpha$$
$$V_{q_0 b}^{\alpha} V_{\gamma}^{\beta} \to V_b^{\alpha} V_{q_0 \gamma}^{\beta} \qquad \beta V_{\gamma}^{\alpha} \to \beta\alpha$$
$$V_{\gamma}^{\beta} V_{q_0 c}^{\alpha} \to V_{q_1 \gamma}^{\beta} V_b^{\alpha} \qquad V_{\gamma}^{\alpha} \beta \to \alpha\beta$$
$$V_{q_1 b}^{\alpha} V_{\gamma}^{\beta} \to V_a^{\alpha} V_{q_2 \gamma}^{\beta} \qquad \Box \to \lambda$$

with $\alpha, \beta \in \Sigma \cup \{\Box\}$ and $\gamma \in \Gamma$. Exercise: derive *abc*.