Теория автоматов и формальных языков

Иерархия Хомского

Автор: Екатерина Вербицкая

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»

19 декабря 2019

В предыдущей серии

- Регулярные языки и конечные автоматы
- Контекстно-свободные языки и магазинные автоматы
- Атрибутные грамматики и магазинные преобразователи

Контекстно-зависимые грамматики

КЗ грамматика: (V_T, V_N, P, S)

- V_T — алфавит терминалов
- V_N — алфавит нетерминалов, $V_T \cap V_N = \varnothing$
- P — конечное множество продукций грамматики вида $\alpha A \beta \to \alpha\gamma\beta : \alpha, \beta \in (V_T \cup V_N)^*, A \in V_N, \gamma \in (V_T \cup V_N)^+$
  - α, β — контекст
- $S \in V_N$ — стартовый нетерминал

Язык: $\{\omega \in V_T^* \mid S \xRightarrow{*} \omega\}$

Неукорачивающие грамматики

Неукорачивающая: (V_T, V_N, P, S)

- V_T — алфавит терминалов
- V_N — алфавит нетерминалов, $V_T \cap V_N = \varnothing$
- P — конечное множество продукций грамматики вида $\alpha \to \beta : \alpha, \beta \in (V_T \cup V_N)^+, 1 \leq |\alpha| \leq |\beta|$
- $S \in V_N$ — стартовый нетерминал

Язык: $\{\omega \in V_T^* \mid S \overset{*}{\Rightarrow} \omega\}$

Эквивалентность КЗ и неукорачивающих грамматик

Теорема

Контекстно-зависимые и неукорачивающие грамматики задают один и тот же класс языков

Доказательство

$\Rightarrow$: *Любая КЗ-грамматика является неукорачивающей*

Эквивалентность КЗ и неукорачивающих грамматик

Доказательство

$\Leftarrow$: *Преобразуем правила неукорачивающей грам. к виду* $\alpha A\beta \to \alpha\gamma\beta$

- *Заменим все вхождения терминалов в правило на нетерминалы, добавим соответствующие правила (вида* $Z \to a, Z \in V_N, a \in V_T$*)*
- *Теперь все правила имеют вид*
  $X_1X_2\dots X_m \to Y_1Y_2\dots Y_{m+q}, m > 0, q \geq 0, X_i, Y_j \in V_N$
- *Такие правила эквивалентны группе правил (требуемого вида):*
  - $X_1X_2\dots X_m \to A_1X_2\dots X_m$
  - $A_1X_2\dots X_m \to A_1A_2\dots X_m$
  - $\dots$
  - $A_1A_2\dots X_m \to A_1A_2\dots A_m$
  - $A_1A_2\dots A_m \to Y_1A_2\dots A_m$
  - $Y_1A_2\dots A_m \to Y_1Y_2\dots A_m$
  - $\dots$
  - $Y_1Y_2\dots A_{m-1}A_m \to Y_1Y_2\dots Y_{m-1}Y_m$
  - $Y_1Y_2\dots Y_{m-1}A_m \to Y_1Y_2\dots Y_{m-1}Y_mY_{m+1}\dots Y_{m+q}$

Пример: язык $\{a^n b^n c^n \mid n \geq 0\}$

Неукорачивающая грамматика:

$$\begin{array}{rcl} S & \to & abc \mid aSQ \\ bQc & \to & bbcc \\ cQ & \to & Qc \end{array}$$

Пример: язык $\{a^n b^n c^n \mid n \geq 0\}$

Неукорачивающая грамматика:

$$\begin{aligned} S &\rightarrow abc \mid aSQ \\ bQc &\rightarrow bbcc \\ cQ &\rightarrow Qc \end{aligned}$$

$S \Rightarrow aSQ \Rightarrow aabcQ \Rightarrow aabQc \Rightarrow aabbcc$

Пример: язык $\{a^n b^n c^n \mid n \geq 0\}$

Неукорачивающая грамматика:

$$\begin{aligned} S &\to abc \mid aSQ \\ bQc &\to bbcc \\ cQ &\to Qc \end{aligned}$$

$S \Rightarrow aSQ \Rightarrow aabcQ \Rightarrow aabQc \Rightarrow aabbcc$

Преобразуем в КЗ-грамматику:

Пример: язык $\{a^n b^n c^n \mid n \geq 0\}$

Неукорачивающая грамматика:

$$\begin{array}{rcl} S & \to & abc \mid aSQ \\ bQc & \to & bbcc \\ cQ & \to & Qc \end{array}$$

$S \Rightarrow aSQ \Rightarrow aabcQ \Rightarrow aabQc \Rightarrow aabbcc$

Преобразуем в КЗ-грамматику:

- $cQ \to Qc$
  - $ZQ \to QZ$
  - $Z \to c$

Пример: язык $\{a^n b^n c^n \mid n \geq 0\}$

Неукорачивающая грамматика:

$$\begin{array}{rcl} S & \to & abc \mid aSQ \\ bQc & \to & bbcc \\ cQ & \to & Qc \end{array}$$

$S \Rightarrow aSQ \Rightarrow aabcQ \Rightarrow aabQc \Rightarrow aabbcc$

Преобразуем в КЗ-грамматику:

- $cQ \to Qc$
  - $ZQ \to QZ$
  - $Z \to c$
- $ZQ \to QZ$
  - $ZQ \to AQ$
  - $AQ \to AB$
  - $AB \to QB$
  - $QB \to QZ$

Рекурсивность КЗ-грамматик

Грамматика **рекурсивна**, если существует алгоритм, определяющий, выводится ли данная строка в данной грамматике

Теорема

Контекстно-зависимые грамматики рекурсивны

Доказательство

Действуем в предположении, что в грамматике нет правила $S \to \varepsilon$
Строим алгоритм, проверяющий, выводится ли в грамматике $\omega \in V_T^+$
Определим множества $T_m = \{\alpha \in (V_T \cup V_N)^+ \mid S \stackrel{i}{\Rightarrow} \alpha, i \leq m, |\alpha| \leq n\}$
$T_0 = \{S\}$; $T_m = T_{m-1} \cup \{\alpha \mid \beta \Rightarrow \alpha, \beta \in T_{m-1}, |\alpha| \leq n\}$; $T_i \subseteq T_{i+1}$
Если $S \stackrel{*}{\Rightarrow} \alpha, |\alpha| \leq n$, *то* $\exists m . \alpha \in T_m$
Последовательно считаем множества T_i, *пока не окажется* $T_m = T_{m-1}$
Количество всех возможных строк заданной длины ограничено, поэтому такая ситуация обязательно настанет
Если $\omega \in T_m$, *то она в языке; иначе — нет. Алгоритм построен*

Линейно-ограниченные автоматы

Линейно-ограниченный автомат — недетерминированная одноленточная МТ, которая никогда не покидает те ячейки, в которых размещен ее вход.

Формально: $(Q, \Sigma, \Gamma, \delta, q_0, F)$

- Q — конечное множество состояний
- $q_0 \in Q$ — стартовое состояние
- $F \subseteq Q$ — множество конечных состояний
- Γ — алфавит допустимых символов ленты
- $\Sigma \subseteq \Gamma$ — входной алфавит
  - $¢ \in \Sigma$ — маркер начала строки
  - $\$ \in \Sigma$ — маркер конца строки
  - Маркеры нельзя перезаписывать или писать на место символов входной строки
- $\delta : Q \times \Gamma \to 2^{Q \times \Gamma \times \{-1, +1\}}$ — отображение перехода

Конфигурация и шаг

- Конфигурация: $(p, A_0, A_1, \ldots, A_{n+1}, i)$
  - $p \in Q$ — текущее состояние автомата
  - $A_0 = ¢$ — маркер начала строки
  - $A_{n+1} = \$$ — маркер конца строки
  - $A_1 \ldots A_n : A_j \in \Gamma$ — содержимое ленты
  - $i \in \mathbb{Z}, 0 \le i \le n+1$ — позиция головки
- Отношение $\vdash$ на конфигурациях (шаг)
  - $\forall (p, A, -1) \in \delta(q, A_i), i > 0$, верно $(q, A_0, A_1, \ldots, A_{n+1}, i) \vdash (p, A_0, A_1, \ldots A_{i-1}, A, A_{i+1}, \ldots, A_{n+1}, i-1)$
  - $\forall (p, A, +1) \in \delta(q, A_i), i < n+1$, верно $(q, A_0, A_1, \ldots, A_{n+1}, i) \vdash (p, A_0, A_1, \ldots A_{i-1}, A, A_{i+1}, \ldots, A_{n+1}, i+1)$
- Языком, принимаемым линейно-ограниченным автоматом, называется
$$\{\omega \in (\Sigma \setminus \{¢, \$\})^* \mid (q_0, ¢\omega\$, 0) \vdash^* (q, ¢\alpha\$, i), \\ q \in F, \alpha \in \Gamma^*, 0 \le i \le n+1, |\omega| = n\}$$

Линейно-ограниченные автоматы и КЗ-языки

Теорема

Для любого КЗ-языка существует линейно-ограниченный автомат, принимающий его

Доказательство

Работаем с двудорожечной МТ: на первой ленте записано входное слово, вторая используется для вывода.
Записываем на вторую дорожку символ S.
На каждом шаге выбираем недетерминированно правило $\alpha \to \beta$, такое что α — подстрока строки, записанной на второй дорожке. Заменяем α на β, сдвигая символы справа от α, если необходимо.
Если на каком-то шаге вышли за пределы длины слова — провал.
Если удалось породить терминальную цепочку, сравниваем ее с входной.
Если совпала — успех, иначе — повторяем процесс

Линейно-ограниченные автоматы и КЗ-языки

Теорема

Если язык принимается линейно-ограниченным автоматом, он является контекстно-зависимым

КЗ-языки и рекурсивные множества

Теорема

Существуют рекурсивные множества, не являющиеся КЗ-языками

Иерархия Хомского: грамматики

Грамматика: V_T, V_N, P, S (обозначим $V = V_T \cup V_N$). В зависимости от вида правил в P, выделяют разные классы грамматик:

- Типа 0: $\alpha \to \beta, \alpha = \gamma A\delta, A \in V_N, \gamma, \delta, \beta \in V^*$
- Типа 1: $\alpha A\beta \to \alpha\gamma\beta, \alpha, \beta \in V^*, A \in V_N, \gamma \in V^+$
  - Или $\alpha \to \beta : 1 \leq |\alpha| \leq \beta$
- Типа 2: $A \to \alpha, A \in V_N, \alpha \in V^*$
- Типа 3: $A \to a$, или $A \to aB$, или $A \to \varepsilon$; $A, B \in V_N, a \in V_T$
  - Или $A \to a$, или $A \to Ba$, или $A \to \varepsilon$; $A, B \in V_N, a \in V_T$

Очевидным образом классы грамматик вкладываются друг в друга

Иерархия Хомского: грамматики, языки, распознаватели

Грамматики	Языки	Распознаватели
Типа 0	неограниченные	машины Тьюринга
Типа 1	контекстно-зависимые	линейно-ограниченные автоматы
Типа 2	контекстно-свободные	магазинные автоматы
Типа 3	регулярные	конечные автоматы