Теория автоматов и формальных языков

Контекстно-свободные языки: нисходящий анализ

Лектор: Екатерина Вербицкая

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»

25 октября 2016г.

В предыдущей серии

- Восходящий и нисходящий синтаксический анализ
- LL-грамматики

Нисходящий синтаксический анализ

- Top-down parsing
- Начинаем разбирать со стартового нетерминала, применяем правила грамматики, пока не получим строку
  - С откатом ([full] backtracking)
  - Без отката (without backtracking)

Нисходящий синтаксический анализ с откатом

- Метод грубой силы, bruteforce
- Перебираем все возможные варианты разбора, если что-то пошло не так — возвращаемся к началу и пробуем снова

Bruteforce parsing: пример

$$
\begin{array}{rcccl}
S & \rightarrow & aAd & | & aB \\
A & \rightarrow & b & | & c \\
B & \rightarrow & ccd & | & ddc
\end{array}
$$

$\omega = addc$

Bruteforce parsing: пример

$$\begin{array}{ccccc} S & \to & aAd & | & aB \\ A & \to & b & | & c \\ B & \to & ccd & | & ddc \end{array}$$

$\omega = addc$
S

Bruteforce parsing: пример

$$\begin{array}{ccccc} S & \rightarrow & aAd & \mid & aB \\ A & \rightarrow & b & \mid & c \\ B & \rightarrow & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd$

Bruteforce parsing: пример

$$
\begin{array}{rclcl}
S & \rightarrow & aAd & | & aB \\
A & \rightarrow & b & | & c \\
B & \rightarrow & ccd & | & ddc
\end{array}
$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \to & aAd & \mid & aB \\ A & \to & b & \mid & c \\ B & \to & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

Bruteforce parsing: пример

$$
\begin{array}{lcccc}
S & \rightarrow & aAd & | & aB \\
A & \rightarrow & b & | & c \\
B & \rightarrow & ccd & | & ddc
\end{array}
$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd$

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \to & aAd & \mid & aB \\ A & \to & b & \mid & c \\ B & \to & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$

Bruteforce parsing: пример

$$\begin{array}{lcccc} S & \rightarrow & aAd & \mid & aB \\ A & \rightarrow & b & \mid & c \\ B & \rightarrow & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

Bruteforce parsing: пример

$$\begin{array}{lcccc} S & \rightarrow & aAd & \mid & aB \\ A & \rightarrow & b & \mid & c \\ B & \rightarrow & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

S

Bruteforce parsing: пример

$$
\begin{array}{lcccl}
S & \rightarrow & aAd & | & aB \\
A & \rightarrow & b & | & c \\
B & \rightarrow & ccd & | & ddc
\end{array}
$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

$S \Rightarrow aB$

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \to & aAd & \mid & aB \\ A & \to & b & \mid & c \\ B & \to & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

$S \Rightarrow aB \Rightarrow accd$

Bruteforce parsing: пример

$$\begin{array}{lcccl} S & \to & aAd & \mid & aB \\ A & \to & b & \mid & c \\ B & \to & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

$S \Rightarrow aB \Rightarrow accd$ — не подходит, откатываемся

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \rightarrow & aAd & \mid & aB \\ A & \rightarrow & b & \mid & c \\ B & \rightarrow & ccd & \mid & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

$S \Rightarrow aB \Rightarrow accd$ — не подходит, откатываемся

$S \Rightarrow aB$

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \rightarrow & aAd & | & aB \\ A & \rightarrow & b & | & c \\ B & \rightarrow & ccd & | & ddc \end{array}$$

$\omega = addc$

$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся

$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся

$S \Rightarrow aB \Rightarrow accd$ — не подходит, откатываемся

$S \Rightarrow aB \Rightarrow addc$

Bruteforce parsing: пример

$$\begin{array}{rcccl} S & \rightarrow & aAd & \mid & aB \\ A & \rightarrow & b & \mid & c \\ B & \rightarrow & ccd & \mid & ddc \end{array}$$

$\omega = addc$
$S \Rightarrow aAd \Rightarrow abd$ — не подходит, откатываемся
$S \Rightarrow aAd \Rightarrow acd$ — не подходит, откатываемся
$S \Rightarrow aB \Rightarrow accd$ — не подходит, откатываемся
$S \Rightarrow aB \Rightarrow addc$ — ура!

Проблема: ну очень уж долго работает: экспоненциальное время!

Нисходящий синтаксический анализ без отката

- Рекурсивный спуск (recursive descent parsing)
  - Для каждого нетерминала написана функция
  - Функции для нетерминалов рекурсивно вызывают друг друга

Рекурсивный спуск: пример

Код

Нисходящий синтаксический анализ без отката: LL(k)

- Идея: откат запрещен, но разрешен предпросмотр
- По (нескольким) следующим терминалам принять решение о том, какую продукцию использовать
- Как и предыдущие 2 подхода не может обрабатывать леворекурсивные правила грамматики
- Достаточно хорош для используемых на практике языков

Леворекурсивные правила грамматики

- Явная (непосредственная) левая рекурсия
  - $A \to A\beta$
- Неявная левая рекурсия
  - $A \to \alpha A\beta$, $\alpha \overset{*}{\Rightarrow} \varepsilon$
- Взаимная рекурсия
  - $A \to \alpha B\beta$, $B \to \gamma A\delta$, $\alpha \overset{*}{\Rightarrow} \varepsilon, \gamma \overset{*}{\Rightarrow} \varepsilon$

Избавление от левой рекурсии

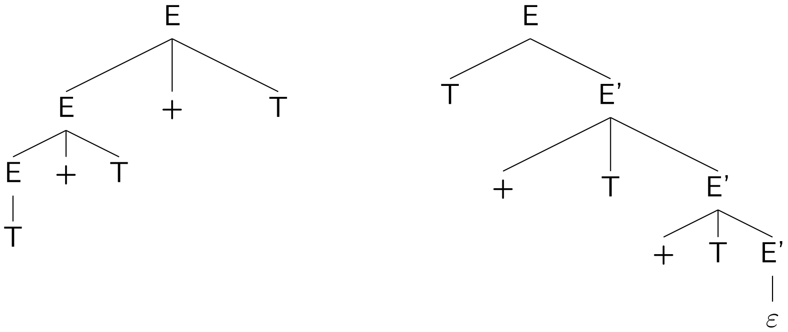
- $A \to A\alpha \,|\, \beta \Leftrightarrow A \to \beta A', \; A' \to \varepsilon \,|\, \alpha A'$

Избавление от левой рекурсии

- $A \to A\alpha \,|\, \beta \Leftrightarrow A \to \beta A',\ A' \to \varepsilon \,|\, \alpha A'$
- $E \to E + T \,|\, T \Leftrightarrow E \to TE',\ E' \to \varepsilon \,|\, +TE'$

Избавление от левой рекурсии

- $A \to A\alpha \,|\, \beta \Leftrightarrow A \to \beta A',\ A' \to \varepsilon \,|\, \alpha A'$
- $E \to E + T \,|\, T \Leftrightarrow E \to TE',\ E' \to \varepsilon \,|\, + TE'$



Избавление от левой рекурсии: более общий случай

- $A \to A\alpha_1 \,|\, A\alpha_2 \,|\, \ldots \,|\, A\alpha_n \,|\, \beta_1 \,|\, \beta_2 \,|\, \ldots \,|\, \beta_k$

- $A \to \beta_1 A' \,|\, \beta_2 A' \,|\, \ldots \,|\, \beta_k A'$
- $A' \to \varepsilon \,|\, \alpha_1 A' \,|\, \alpha_2 A' \,|\, \ldots \,|\, \alpha_n A'$

Избавление от взаимной левой рекурсии

- Избавляемся от ε-продукций
- Упорядочиваем правила по индексу нетерминала
- Добиваемся того, чтобы не было правил вида $A_i \to A_j\alpha, j \leq i$
  - Перебираем все A_i
  - Перебираем все $A_j, 1 \leq j < i$
  - Для каждого правила $p : A_i \to A_j\gamma$
    - Удалить правило p
    - Для каждого правила $A_j \to \delta_1 \mid \ldots \mid \delta_k$ Добавить правила $A_i \to \delta_l$
  - Устранить непосредственную левую рекурсию для A_i

Левая факторизация грамматики

- Выделяем наибольший общий префикс продукций
  $A \to \alpha\beta \,|\, \alpha\gamma \Rightarrow A \to \alpha A',\ A' \to \beta \,|\, \gamma$

Пример

$$
\begin{array}{rcl}
S & \to & aSSbS \\
 & | & aSaSb \\
 & | & abb \\
 & | & b
\end{array}
$$

Пример

$$
\begin{array}{rcl}
S & \to & aSSbS \\
 & | & aSaSb \\
 & | & abb \\
 & | & b
\end{array}
$$

$$
\begin{array}{rcl}
S & \to & aS' \\
 & | & b \\
S' & \to & SSbS \\
 & | & SaSb \\
 & | & bb
\end{array}
$$

Пример

$$
\begin{array}{rcl}
S & \rightarrow & aSSbS \\
& | & aSaSb \\
& | & abb \\
& | & b
\end{array}
$$

$$
\begin{array}{rcl}
S & \rightarrow & aS' \\
& | & b \\
S' & \rightarrow & SSbS \\
& | & SaSb \\
& | & bb
\end{array}
$$

$$
\begin{array}{rcl}
S & \rightarrow & aS' \mid b \\
S' & \rightarrow & SS'' \mid bb \\
S'' & \rightarrow & SbS \mid aSb
\end{array}
$$

Множество FIRST

- Множество символов, которые могут появиться первыми во время вывода из данной сентенциальной формы
- $FIRST(a\alpha) = \{a\}$, $a \in V_T, \alpha \in (V_T \cup V_N)^*$
- $FIRST(\varepsilon) = \{\varepsilon\}$
- $FIRST(\alpha\beta) = FIRST(\alpha) \cup (FIRST(\beta), if \varepsilon \in FIRST(\alpha))$
- $FIRST(S) = FIRST(\alpha) \cup FIRST(\beta), S \to \alpha \mid \beta$

Множество FIRST: пример

$$
\begin{array}{rcl}
S & \rightarrow & aS' \\
S' & \rightarrow & AbBS' \mid \varepsilon \\
A & \rightarrow & aA' \mid \varepsilon \\
A' & \rightarrow & b \mid a \\
B & \rightarrow & c \mid \varepsilon
\end{array}
$$

Множество FIRST: пример

$$
\begin{array}{rcl}
S & \rightarrow & aS' \\
S' & \rightarrow & AbBS' \mid \varepsilon \\
A & \rightarrow & aA' \mid \varepsilon \\
A' & \rightarrow & b \mid a \\
B & \rightarrow & c \mid \varepsilon
\end{array}
$$

- $FIRST(S) = \{a\}$

Множество FIRST: пример

$$
\begin{aligned}
S &\rightarrow aS' \\
S' &\rightarrow AbBS' \mid \varepsilon \\
A &\rightarrow aA' \mid \varepsilon \\
A' &\rightarrow b \mid a \\
B &\rightarrow c \mid \varepsilon
\end{aligned}
$$

- $FIRST(S) = \{a\}$
- $FIRST(A) = \{a, \varepsilon\}$

Множество FIRST: пример

$$\begin{array}{rcl} S & \to & aS' \\ S' & \to & AbBS' \mid \varepsilon \\ A & \to & aA' \mid \varepsilon \\ A' & \to & b \mid a \\ B & \to & c \mid \varepsilon \end{array}$$

- $FIRST(S) = \{a\}$
- $FIRST(A) = \{a, \varepsilon\}$
- $FIRST(A') = \{a, b\}$

Множество FIRST: пример

$$
\begin{array}{rcl}
S & \to & aS' \\
S' & \to & AbBS' \mid \varepsilon \\
A & \to & aA' \mid \varepsilon \\
A' & \to & b \mid a \\
B & \to & c \mid \varepsilon
\end{array}
$$

- $FIRST(S) = \{a\}$
- $FIRST(A) = \{a, \varepsilon\}$
- $FIRST(A') = \{a, b\}$
- $FIRST(B) = \{c, \varepsilon\}$

Множество FIRST: пример

$$
\begin{array}{rcl}
S & \to & aS' \\
S' & \to & AbBS' \mid \varepsilon \\
A & \to & aA' \mid \varepsilon \\
A' & \to & b \mid a \\
B & \to & c \mid \varepsilon
\end{array}
$$

- $FIRST(S) = \{a\}$
- $FIRST(A) = \{a, \varepsilon\}$
- $FIRST(A') = \{a, b\}$
- $FIRST(B) = \{c, \varepsilon\}$
- $FIRST(S') = \{a, b, \varepsilon\}$

Множество FOLLOW

- Множество символов, которые могут появиться в некотором выводе сразу после данной сентенциальной формы
- Положим $FOLLOW(X) = \varnothing$
- Если X — стартовый нетерминал, $FOLLOW(X) = FOLLOW(X) \cup \{\$\}$ — символ конца строки
- Для всех правил вида $A \to \alpha X \beta$, $FOLLOW(X) = FOLLOW(X) \cup (FIRST(\beta) \setminus \{\varepsilon\})$
- Для всех правил вида $A \to \alpha X$ и $A \to \alpha X \beta$, где $\varepsilon \in FIRST(\beta)$, $FOLLOW(X) = FOLLOW(X) \cup FOLLOW(A)$
- Повторять последние 2 пункта, пока можно что-то добавлять

Множество FOLLOW: пример

$$
\begin{array}{rcl}
S & \rightarrow & aS' \\
S' & \rightarrow & AbBS' \mid \varepsilon \\
A & \rightarrow & aA' \mid \varepsilon \\
A' & \rightarrow & b \mid a \\
B & \rightarrow & c \mid \varepsilon
\end{array}
$$

Множество FOLLOW: пример

$$\begin{array}{rcl} S & \rightarrow & aS' \\ S' & \rightarrow & AbBS' \mid \varepsilon \\ A & \rightarrow & aA' \mid \varepsilon \\ A' & \rightarrow & b \mid a \\ B & \rightarrow & c \mid \varepsilon \end{array}$$

- $FOLLOW(S) = \{\$\}$

Множество FOLLOW: пример

$$\begin{array}{rcl} S & \rightarrow & aS' \\ S' & \rightarrow & AbBS' \mid \varepsilon \\ A & \rightarrow & aA' \mid \varepsilon \\ A' & \rightarrow & b \mid a \\ B & \rightarrow & c \mid \varepsilon \end{array}$$

- $FOLLOW(S) = \{\$\}$
- $FOLLOW(S') = \{\$\}\,(S \rightarrow aS')$

Множество FOLLOW: пример

$$\begin{array}{rcl} S & \to & aS' \\ S' & \to & AbBS' \mid \varepsilon \\ A & \to & aA' \mid \varepsilon \\ A' & \to & b \mid a \\ B & \to & c \mid \varepsilon \end{array}$$

- $FOLLOW(S) = \{\$\}$
- $FOLLOW(S') = \{\$\}\,(S \to aS')$
- $FOLLOW(A) = \{b\}\,(S' \to AbBS')$

Множество FOLLOW: пример

$$
\begin{aligned}
S &\rightarrow aS' \\
S' &\rightarrow AbBS' \mid \varepsilon \\
A &\rightarrow aA' \mid \varepsilon \\
A' &\rightarrow b \mid a \\
B &\rightarrow c \mid \varepsilon
\end{aligned}
$$

- $FOLLOW(S) = \{\$\}$
- $FOLLOW(S') = \{\$\} \, (S \rightarrow aS')$
- $FOLLOW(A) = \{b\} \, (S' \rightarrow AbBS')$
- $FOLLOW(A') = \{b\} \, (A \rightarrow aA')$

Множество FOLLOW: пример

$$
\begin{array}{rcl}
S & \to & aS' \\
S' & \to & AbBS' \mid \varepsilon \\
A & \to & aA' \mid \varepsilon \\
A' & \to & b \mid a \\
B & \to & c \mid \varepsilon
\end{array}
$$

- $FOLLOW(S) = \{\$\}$
- $FOLLOW(S') = \{\$\}\,(S \to aS')$
- $FOLLOW(A) = \{b\}\,(S' \to AbBS')$
- $FOLLOW(A') = \{b\}\,(A \to aA')$
- $FOLLOW(B) = \{a, b, \$\}\,(S' \to AbBS', \varepsilon \in FIRST(S'))$

LL(1)-анализ

- Нисходящий синтаксический анализ с предпросмотром одного символа
- Читает вход слева направо (L: left-to-right), строит левый вывод в грамматике (L: leftmost)
- Состоит из:
  - Входного буфера (откуда читается входная строка)
  - Стека (для промежуточных данных)
  - Таблицы анализатора (управляет процессом разбора)
- Работает за $O(n)$, где n — длина входной строки

$$S \to (S) \mid \varepsilon$$

Размещаем продукции в таблице (по горизонтали — нетерминалы; по вертикали — терминалы + \$)

- Продукции вида $A \to \alpha$ — в ячейки (A, a), где $a \in FIRST(A)$
- Продукции вида $A \to \varepsilon$ — в ячейки (A, a), где $a \in FOLLOW(A)$

N	FIRST	FOLLOW	(	)	\$
S					

Таблица LL(1)-анализатора

$$S \to (S) \mid \varepsilon$$

Размещаем продукции в таблице (по горизонтали — нетерминалы; по вертикали — терминалы + \$)

- Продукции вида $A \to \alpha$ — в ячейки (A, a), где $a \in FIRST(A)$
- Продукции вида $A \to \varepsilon$ — в ячейки (A, a), где $a \in FOLLOW(A)$

N	FIRST	FOLLOW	(	)	\$
S	$\{(, \varepsilon\}$	$\{), \$\}$			

Таблица LL(1)-анализатора

$$S \to (S) \mid \varepsilon$$

Размещаем продукции в таблице (по горизонтали — нетерминалы; по вертикали — терминалы + \$)

- Продукции вида $A \to \alpha$ — в ячейки (A, a), где $a \in FIRST(A)$
- Продукции вида $A \to \varepsilon$ — в ячейки (A, a), где $a \in FOLLOW(A)$

N	FIRST	FOLLOW	(	)	\$
S	$\{(, \varepsilon\}$	$\{), \$\}$	$S \to (S)$		

Таблица LL(1)-анализатора

$$S \to (S) \mid \varepsilon$$

Размещаем продукции в таблице (по горизонтали — нетерминалы; по вертикали — терминалы + $)

- Продукции вида $A \to \alpha$ — в ячейки (A, a), где $a \in FIRST(A)$
- Продукции вида $A \to \varepsilon$ — в ячейки (A, a), где $a \in FOLLOW(A)$

N	FIRST	FOLLOW	(	)	$
S	$\{(, \varepsilon\}$	$\{), \$\}$	$S \to (S)$	$S \to \varepsilon$	$S \to \varepsilon$

Синтаксический анализ (доска)

$$S \to (S) \mid \varepsilon$$

N	FIRST	FOLLOW	(	)	\$
S	$\{(, \varepsilon\}$	$\{), \$\}$	$S \to (S)$	$S \to \varepsilon$	$S \to \varepsilon$

$\omega = (())\$$
Стек: $\$, S,), S, (,), S, ($

Когда LL-анализ не возможен

- Леворекурсивные правила
- Когда при построении таблицы в одну ячейку нужно записать больше одной записи
  - FIRST-FIRST конфликт
    - $A \rightarrow \alpha \mid \beta, FIRST(\alpha) \cap FIRST(\beta) \neq \varnothing$
    - $E \rightarrow T + E \mid T * E$
  - FIRST-FOLLOW конфликт
    - $FIRST(A) \cap FOLLOW(A) \neq \varnothing$
    - $S \rightarrow Aab, A \rightarrow a \mid \varepsilon$
- Как с этим бороться?
  - Избавиться от левой рекурсии
  - Избавиться от недетерминизма
  - Факторизовать грамматику
  - Использовать аннотации (если есть)
  - Переписать грамматику
  - Использовать более одного символа предпросмотра