ITC: Introdução à Teoria da Computação

Marcos Castilho

DInf/UFPR

12 de julho de 2021

Gramáticas Irrestritas

Definição: Uma Gramática Irrestrita é uma quádrupla (V, Σ, P, S) onde V é um conjunto finito de variáveis, Σ é um conjunto finito de símbolos terminais, P é um conjunto de regras de produção e S é um elemento de V. As produções em P têm a forma:

$$u \rightarrow v$$

onde:

- $u \in (V \cup \Sigma)^+$
- $v \in (V \cup \Sigma)^*$
- $V \cap \Sigma = \emptyset$

Exemplo

A gramática irrestrita $G = (V, \Sigma, P, S)$:

$$
\begin{aligned}
&V = \{S, A, C\} \\
&\Sigma = \{a, b, c\} \\
&S \to aAbc \mid \lambda \\
&A \to aAbC \mid \lambda \\
&Cb \to bC \\
&Cc \to cc
\end{aligned}
$$

Gera $L = \{a^i b^i c^i \mid i \geq 0\}$.

Exemplo, continuação

Para gerar $a^i b^i c^i, i \geq 0$:

$$S \Rightarrow aAbc \Rightarrow^{i-1} a^i A(bc)^{i-1}bc \Rightarrow a^i(bC)^{i-1}bc$$

A regra $Cb \rightarrow bC$ leva o c até o final e em seguida, a regra $Cc \rightarrow cc$ transforma os C em c.

Teorema

Seja $G = (V, \Sigma, P, S)$ uma gramática irrestrita. Então $L(G)$ é uma linguagem recursivamente enumerável.

Prova

Construímos uma máquina de Turing não determinística com 3 fitas.

- A fita 1 terá uma entrada $p \in \Sigma^*$.
- M será projetada para simular as derivações de G
- Uma representação das regras de G estará na fita 2
  $u \to v \Leftrightarrow u\#v$
- Separamos as regras com $\#\#$
- As derivações de G são simuladas na fita 3

A seguir a computação de M é mostrada:

Prova

1. S é escrito na primeira posição da fita 3
2. As regras de G são escritas na fita 2
3. Uma regra $u\#v$ é escolhida na fita 2
4. Uma instância da palavra u é escolhida na fita 3, se existir uma. Senão a computação termina em estado não final
5. A palavra u é trocada por v na fita 3
6. Se as palavras das fitas 3 e 1 casam, a computação termina em um estado final
7. Repetir passos 3-7.

Se $p \in L(G)$ uma computação não determinística vai derivar p.

Exemplo

Seja $L = \{a^i b^i c^i \mid i \geq 0\}$:

$$\begin{aligned}
&V = \{S, A, C\} \\
&\Sigma = \{a, b, c\} \\
&S \to aAbc \mid \lambda \\
&A \to aAbC \mid \lambda \\
&Cb \to bC \\
&Cc \to cc
\end{aligned}$$

Fita 2:
$BS\#aAbc\#\#S\#\#\#A\#aAbC\#\#A\#\#\#Cb\#bC\#\#Cc\#ccB$

Teorema

Seja L uma linguagem recursivamente enumerável. Então existe uma gramática irrestrita G tal que $L = L(G)$.

Prova

Ver páginas 301-302 do livro do Sudkamp. A ideia é simular as transições de uma máquina de Turing usando uma gramática irrestrita.

Gramáticas Sensíveis ao Contexto

Definição: Uma Gramática $G = (V, \Sigma, P, S)$ é chamada de *sensível ao contexto* se cada produção em P têm a forma:

$$u \rightarrow v$$

onde:

- $u \in (V \cup \Sigma)^+$
- $v \in (V \cup \Sigma)^*$
- $|u| \leq |v|$

Estas regras são chamadas *monotônicas*.

Gramáticas Sensíveis ao Contexto

A linguagem gerada por uma gramática sensível ao contexto é chamada de *Linguagem Senvível ao Contexto*.

Obs.: Originalmente, as regras de uma GSC tinham a forma $uAv \rightarrow uwv$, onde $A \in V, w \in (V \cup \Sigma)^+$ e $u, v \in (V \cup \Sigma)^*$, significando que *$A$ pode ser trocada por w apenas no contexto u, v*.

Teorema

Toda linguagem sensível ao contexto é recursiva.

Prova

Ver páginas 305-306 no livro do Sudkamp.

Autômatos Limitados Linearmente

Definição: Um *Autômato Limitado Linearmente* é uma estrutura $M = (Q, \Sigma, \Gamma, \delta, q_0, <, >, F)$, onde $Q, \Sigma, \Gamma, \delta, q_0$ e F são como no caso da máquina de Turing não determinística. Os símbolos $<$ e $>$ são elementos que não estão em Σ.

Autômatos Limitados Linearmente

- A configuração inicial de uma computação é $q_0 < w >$, o que requer $|w| + 2$ posições da fita;
- $<$ e $>$ são escritos na fita, mas não são parte da entrada;
- As marcas $<$ e $>$ podem ser lidas mas não podem ser apagadas;
- As transições que lêem o $<$ devem fazer o cabeçote se mover para a direita, e o mesmo com relação ao $>$ que fazem o cabeçote ser mover para a esquerda;
- Uma palavra $w \in (\Sigma - \{<, >\})^*$ é aceita por um ALL se uma computação com entrada $< w >$ para em um estado final.

Teorema

Seja L uma Linguagem Sensível ao Contexto. Então existe um Autômato Limitado Linearmente M com $L = L(M)$.

Prova

Páginas 307-308 do livro do Sudkamp.

Teorema

Seja L uma Linguagem aceita por um Autômato Limitado Linearmente. Então $L - \{\lambda\}$ é uma Linguagem Sensível ao Contexto.

Prova

Páginas 308-309 do livro do Sudkamp.

A Hierarquia de Chomsky

Chomsky numerou as quatro famílias de gramáticas, definindo assim a hierarquia:

- Tipo 0: Gramáticas Irrestritas
- Tipo 1: Gramáticas Sensíveis ao Contexto
- Tipo 2: Gramáticas Livres de Contexto
- Tipo 3: Gramáticas Regulares

A hierarquia de Chomsky

- Quanto maior o número, maiores são as restrições nas regras;
- Toda Gramática tipo i é também uma de tipo $i - 1$, observando-se atenção com relação à λ nos tipos 1 e 2;
- As inclusões inversas são próprias;
- Se incluirmos os diversos resultados vistos no curso temos:

Resumo

Gramáticas	Linguagens	Máquinas
Tipo 0	Recursivamente Enumeráveis	Máquinas de Turing
Tipo 1	Sensíveis ao Contexto	Autômatos Limitados Linearmente
Tipo 2	Livre de Contexto	Autômatos com Pilha
Tipo 3	Regulares	Autômatos Finitos

Observação final

É possível generalizar o conceito de Autômatos com Pilha para que ele tenha duas pilhas. Algumas observações interessantes seguem:

- Não adianta colocar mais do que duas pilhas;
- Esta máquina aceita as linguagens recursivamente enumeráveis.

Exercícios

1. Construa gramáticas irrestritas para as seguintes linguagens:
   1.1 $\{a^i b^j a^i b^j \mid i, j \geq 0\}$
   1.2 $\{a^i b^i c^i d^i \mid i \geq 0\}$
   1.3 $\{www \mid w \in \{a, b\}^*\}$
2. Prove que as linguagens recursivamente enumeráveis são fechadas sob união.

Exercícios, continuação

3. Seja G a gramática:

$$\begin{aligned} S &\to SBA \mid a \\ BA &\to AB \mid \lambda \\ aA &\to aaB \\ B &\to b \end{aligned}$$

   3.1 Dê uma derivação de *aabb*.
   3.2 Qual é a linguagem $L(G)$?
   3.3 Construa uma gramática livre de contexto que gera $L(G)$.

Exercícios, continuação

4. Seja L a linguagem $\{a^i b^{2i} a^i \mid i > 0\}$.
   4.1 Use o lema do bombeamento para linguagens livre de contexto para mostrar que L não é livre de contexto.
   4.2 Construa uma gramática sensível ao contexto G que gera L.
   4.3 Dê uma derivação de $aabbbbaa$ em G.
   4.4 Construa um autômato limitado linearmente que aceita L.
   4.5 Mostre a computação de M com a entrada $aabbbbaa$

Licença

- Slides feitos em LaTeX usando beamer e tikz, editados com vim.
- Licença

  Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.http://creativecommons.org/licenses/by-nc-nd/2.5/br/

  Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.http://creativecommons.org/licenses/by-nc-nd/2.5/br/