Reduction of Scattered Context Generators of Sentences Preceded by Their Leftmost Parses

Jiří Techet Tomáš Masopust Alexander Meduna

Department of Information Systems
Faculty of Information Technology
Brno University of Technology
Božetěchova 2, Brno 61266, Czech Republic

Modern Formal Language Theory, 2007

Scattered Context Grammar

Scattered context grammar (SC grammar)

$G = (V, T, P, S)$, where

V is a finite alphabet

T is a set of terminals, $T \subset V$

S is the start symbol, $S \in V - T$

P is a finite set of productions of the form

$$(A_1, \ldots, A_n) \to (x_1, \ldots, x_n),$$

where $A_1, \ldots, A_n \in V - T$, $x_1, \ldots, x_n \in V^*$

Propagating scattered context grammar (PSC grammar)

- each $(A_1, \ldots, A_n) \to (x_1, \ldots, x_n)$ satisfies $x_1, \ldots, x_n \in V^+$

Derivation Step

Derivation step

For $(A_1, \ldots, A_n) \rightarrow (x_1, \ldots, x_n) \in P$ and

$$u = u_1 A_1 \ldots u_n A_n u_{n+1}$$
$$v = u_1 x_1 \ldots u_n x_n u_{n+1}$$

we write $u \Rightarrow v \; [(A_1, \ldots, A_n) \rightarrow (x_1, \ldots, x_n)]$

Leftmost derivation step

- each $A_i \notin \mathrm{alph}(u_i)$ for all $1 \leq i \leq n$
- $\mathrm{alph}(w)$ denotes the set of all symbols occuring in w

Generated Language

Generated language

$L(G) = \{x \in T^* : S \Rightarrow^* x\}$

Language generated in a leftmost way

- in addition, each step in every successful derivation is leftmost

Generative power

- $\mathscr{L}(SC) = \mathscr{L}(RE)$
- $\mathscr{L}(CF) \subset \mathscr{L}(PSC) \subseteq \mathscr{L}(CS)$

PSC Grammar—Example

Example

SC grammar $G_1 = (\{A, B, C, S, a, b, c\}, \{a, b, c\}, P_1, S)$ with

$$P_1 = \{(S) \to (ABC), (A, B, C) \to (aA, bB, cC), (A, B, C) \to (\varepsilon, \varepsilon, \varepsilon)\}$$

$S \Rightarrow ABC \Rightarrow aAbBcC \Rightarrow aaAbbBccC \Rightarrow aabbcc$
$L(G_1) = \{a^n b^n c^n \ : \ n \geq 0\}$

Example

PSC grammar $G_2 = (\{A, B, C, S, a, b, c\}, \{a, b, c\}, P_2, S)$ with

$$P_2 = \{(S) \to (\varepsilon), (S) \to (ABC), (A, B, C) \to (aA, bB, cC), (A, B, C) \to (a, b, c)\}$$

Production Labels I

- for every grammar G there is a set of production labels
- we denote them $lab(G)$
- each $p \in lab(G)$ uniquely identifies one production
- we write $p : (A_1, \ldots, A_n) \rightarrow (x_1, \ldots, x_n)$

Example

$$G = (\{S, A, B, C, a, b, c\}, \{a, b, c\}, P, S)$$

with

$$lab(G) = \{1, 2, 3\}$$

$$\begin{aligned} P = \{&1 : (S) \rightarrow (ABC), \\ &2 : (A, B, C) \rightarrow (aA, bB, cC), \\ &3 : (A, B, C) \rightarrow (\epsilon, \epsilon, \epsilon)\} \end{aligned}$$

$$L(G) = \{a^n b^n c^n \ : \ n \geq 0\}$$

Production Labels II

- to express that $x \Rightarrow y$ by $p : (A_1, \ldots, A_n) \rightarrow (x_1, \ldots, x_n)$, we write $x \Rightarrow y$ $[p]$

Example

$S \Rightarrow ABC\ [1] \Rightarrow aAbBcC\ [2] \Rightarrow aaAbbBccC\ [2] \Rightarrow aabbcc\ [3]$ in G

- to express that $x \Rightarrow^* y$ by productions labeled with $p_1, \ldots, p_n$, we write $x \Rightarrow^* y$ $[p_1 \ldots p_n]$
- $p_1 \ldots p_n \in lab(G)^*$

Example

$S \Rightarrow^* aabbcc\ [1223]$ in G
$1223 \in lab(G)^*$

Generator of its Sentences Preceded by Their Parses

Parse (Szilard word, control word)

If $S \Rightarrow^* x\,[\rho], x \in T^*, \rho \in lab(G)^*$, then x is a sentence generated by G according to parse ρ

Example

aabbcc is a sentence generated according to parse 1223 in G

Proper generator of its sentences preceded by their parses

G is a proper generator of its sentences preceded by their parses iff
$L(G) = \{x : x = \rho y, y \in (T - lab(G))^*, \rho \in lab(G)^*, S \Rightarrow^* x\,[\rho]\}$

Proper leftmost generator of its sentences preceded by their parses

- in addition, G generates $L(G)$ in a leftmost way

Proper Generator of its Sentences Preceded by Their Parses—Example

Example

$$G = (\{S, A, B, C, a, b, c, 1, 2, 3, \$\}, \{a, b, c, 1, 2, 3\}, P, S)$$

with

$$lab(G) = \{1, 2, 3\},$$

$$\begin{aligned} P = \{1 &: (S) \rightarrow (1\$ABC) \\ 2 &: (\$, A, B, C) \rightarrow (2\$, aA, bB, cC) \\ 3 &: (\$, A, B, C) \rightarrow (3, \epsilon, \epsilon, \epsilon)\} \end{aligned}$$

$S \Rightarrow 1\$ABC\ [1] \Rightarrow 12\$aAbBcC\ [2] \Rightarrow 122\$aaAbbBccC\ [2] \Rightarrow$
$1223aabbcc\ [3]$

$S \Rightarrow^* 1223aabbcc\ [1223]$

$L(G) = \{\rho a^n b^n c^n \ : \ n \geq 0, S \Rightarrow^* \rho a^n b^n c^n\ [\rho], \rho = 12^n3\}$

G is a proper leftmost generator of its sentences preceded by their parses

Theorem 1

Production Length

Let $G = (V, T, P, S)$ be a SC grammar. Set

- $\text{len}((A_1, \ldots, A_n) \to (x_1, \ldots, x_n)) = |A_1 \ldots A_n| = n$
- $\text{len}_{max}(P) = \text{len}(p)$, where $p \in P$ is a production satisfying $\text{len}(p) \geq \text{len}(r)$ for all $r \in P$

Left Quotient

$L_2 \backslash L_1 = \{y \, : \, xy \in L_1, \text{ for some } x \in L_2\}$

Theorem

For every recursively enumerable language L there exists a PSC grammar $G = (V, T, P, S)$ such that G is a proper leftmost generator of its sentences preceded by their parses, G contains no more than six nonterminals, $\text{len}_{max}(P) = 4$, and $L = \text{lab}(G)^+ \backslash L(G) \cap \text{alph}(L)^*$.

Extended Post Correspondence

Extended Post Correspondence (EPC)

$$E = (\{(u_1, v_1), \ldots, (u_r, v_r)\}, (z_{a_1}, \ldots, z_{a_n})),$$

where $T = \{a_1, \ldots, a_n\}$, $u_i, v_i, z_{a_j} \in \{0, 1\}^*$ for each $1 \leq i \leq r$ and $1 \leq j \leq n$

Language of EPC

$$L(E) = \{b_1 \ldots b_k \in T^* \; : \; v_{s_1} \ldots v_{s_l} = u_{s_1} \ldots u_{s_l} z_{b_1} \ldots z_{b_k} \text{ for some } s_1, \ldots, s_l \in \{1, \ldots, r\}, l \geq 1, k \geq 0\}$$

Example

$E = (\{(01, 011), (1, 10), (1, 11)\}, (1_a, 01_b)), T = \{a, b\}$

$0111011 = 011101_b1_a$

$ba \in L(E)$

Power of EPC

Theorem

Every recursively enumerable language can be represented by an EPC.

Representation of a RE language

- let $L \subseteq T^*$, where $T = \{a_1, \ldots, a_n\}$, be a RE language
- L is represented by an EPC

$$E = (D, (z_{a_1}, \ldots, z_{a_n})),$$

where

$$D = \{(u_1, v_1), \ldots, (u_r, v_r)\},$$

$u_i, v_i, z_{a_j} \in \{0, 1\}^*$ for each $1 \leq i \leq r$, $1 \leq j \leq n$

Construction

Basic Idea

0. Represent the RE language L by an EPC
1. Generate strings $z_{a_{j_1}} \dots z_{a_{j_n}}$ and $a_{j_1} \dots a_{j_n}$
2. Generate strings $u_{s_1} \dots u_{s_l}$ and $v_{s_1} \dots v_{s_l}$
3. Verify if $u_{s_1} \dots u_{s_l} z_{a_{j_1}} \dots z_{a_{j_n}} = v_{s_1} \dots v_{s_l}$
4. Replace all auxiliary symbols with labels

$+$ after using a production, add its label to a sentential form

Define the PSC grammar

$$G = (\{S, A, B, 0, 1, \#\} \cup T \cup \mathrm{lab}(G), T \cup \mathrm{lab}(G), P, S),$$

where

$$\begin{aligned}\mathrm{lab}(G) = {} & \{\lfloor 1 \rfloor, \lfloor 3 \rfloor, \lfloor 3_0 \rfloor, \lfloor 3_1 \rfloor, \lfloor 4 \rfloor, \lfloor 4_0 \rfloor, \lfloor 4_1 \rfloor, \lfloor 4_2 \rfloor\} \\ & \cup \{\lfloor 1_a \rfloor : a \in T\} \cup \{\lfloor 2_{u_i v_i} \rfloor, \lfloor 2_{0 u_i v_i} \rfloor : (u_i, v_i) \in D\}\end{aligned}$$

Construction of P, Step 1

Step 1

For each $a \in T$, add

1. $\lfloor 1 \rfloor : (S) \rightarrow (\lfloor 1 \rfloor AA)$
2. $\lfloor 1_a \rfloor : (A, A) \rightarrow (\lfloor 1_a \rfloor A z_a, Aa)$

Example

$$\begin{aligned} S &\Rightarrow \lfloor 1 \rfloor AA \\ &\Rightarrow \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor A z_{a_{j_n}} A a_{j_n} \\ &\quad \vdots \\ &\Rightarrow \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor \dots \lfloor 1_{a_{j_1}} \rfloor A z_{a_{j_1}} \dots z_{a_{j_n}} A a_{j_1} \dots a_{j_n} \end{aligned}$$

Construction of P, Step 2

Step 2

For each $(u_i, v_i) \in D$, $1 \leq i \leq r$, add

1. $\lfloor 2_{u_i v_i} \rfloor : (A, A) \rightarrow (\lfloor 2_{u_i v_i} \rfloor B u_i, B v_i)$
2. $\lfloor 2_{0 u_i v_i} \rfloor : (B, B) \rightarrow (\lfloor 2_{0 u_i v_i} \rfloor B u_i, B v_i)$

Example

$$
\begin{aligned}
& \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor \ldots \lfloor 1_{a_{j_1}} \rfloor A z_{a_{j_1}} \ldots z_{a_{j_n}} A a_{j_1} \ldots a_{j_n} \\
\Rightarrow & \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor \ldots \lfloor 1_{a_{j_1}} \rfloor \lfloor 2_{u_{s_l} v_{s_l}} \rfloor B u_{s_l} z_{a_{j_1}} \ldots z_{a_{j_n}} B v_{s_l} a_{j_1} \ldots a_{j_n} \\
\Rightarrow & \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor \ldots \lfloor 1_{a_{j_1}} \rfloor \lfloor 2_{u_{s_l} v_{s_l}} \rfloor \lfloor 2_{0 u_{s_{l-1}} v_{s_{l-1}}} \rfloor \\
& B u_{s_{l-1}} u_{s_l} z_{a_{j_1}} \ldots z_{a_{j_n}} B v_{s_{l-1}} v_{s_l} a_{j_1} \ldots a_{j_n} \\
& \vdots \\
\Rightarrow & \lfloor 1 \rfloor \lfloor 1_{a_{j_n}} \rfloor \ldots \lfloor 1_{a_{j_1}} \rfloor \lfloor 2_{u_{s_l} v_{s_l}} \rfloor \lfloor 2_{0 u_{s_{l-1}} v_{s_{l-1}}} \rfloor \ldots \lfloor 2_{0 u_{s_1} v_{s_1}} \rfloor \\
& B u_{s_1} \ldots u_{s_l} z_{a_{j_1}} \ldots z_{a_{j_n}} B v_{s_1} \ldots v_{s_l} a_{j_1} \ldots a_{j_n}
\end{aligned}
$$

Construction of P, Step 3

Step 3 $(u_{s_1} \dots u_{s_l} z_{a_{j_1}} \dots z_{a_{j_n}} \overset{?}{=} v_{s_1} \dots v_{s_l})$

Add

1. $\lfloor 3 \rfloor : (B, B) \rightarrow (\lfloor 3 \rfloor A, B)$
2. $\lfloor 3_0 \rfloor : (A, 0, B, 0) \rightarrow (\lfloor 3_0 \rfloor, A, \#, B)$
3. $\lfloor 3_1 \rfloor : (A, 1, B, 1) \rightarrow (\lfloor 3_1 \rfloor, A, \#, B)$

Example

$$\begin{aligned}
& \dots B u_{s_1} \dots u_{s_l} z_{a_{j_1}} \dots z_{a_{j_n}} B v_{s_1} \dots v_{s_l} a_{j_1} \dots a_{j_n} \\
\Rightarrow\ & \dots \lfloor 3 \rfloor A 01 \dots 01 B 01 \dots 01 a_{j_1} \dots a_{j_n} \\
\Rightarrow\ & \dots \lfloor 3 \rfloor \lfloor 3_0 \rfloor A 1 \dots 01 \# B 1 \dots 01 a_{j_1} \dots a_{j_n} \\
& \vdots \\
\Rightarrow\ & \dots \lfloor 3 \rfloor \lfloor 3_0 \rfloor \dots \lfloor 3_1 \rfloor A \# \dots \# B a_{j_1} \dots a_{j_n}
\end{aligned}$$

Construction of P, Step 4

Step 4

Add

1. $\lfloor 4 \rfloor : (A, B) \rightarrow (\lfloor 4 \rfloor B, A)$
2. $\lfloor 4_0 \rfloor : (B, \#) \rightarrow (\lfloor 4_0 \rfloor, B)$
3. $\lfloor 4_1 \rfloor : (B, A) \rightarrow (\lfloor 4_1 \rfloor, B)$
4. $\lfloor 4_2 \rfloor : (B) \rightarrow (\lfloor 4_2 \rfloor)$

Example

$$
\begin{aligned}
& \dots A\# \dots \# B a_{j_1} \dots a_{j_n} \Rightarrow \ \dots \lfloor 4 \rfloor B \# \dots \# A a_{j_1} \dots a_{j_n} \\
\Rightarrow\ & \dots \lfloor 4 \rfloor \lfloor 4_0 \rfloor B \# \dots \# A a_{j_1} \dots a_{j_n} \Rightarrow \ \dots \\
\dots \Rightarrow\ & \dots \lfloor 4 \rfloor \lfloor 4_0 \rfloor \dots \lfloor 4_0 \rfloor B A a_{j_1} \dots a_{j_n} \\
\Rightarrow\ & \dots \lfloor 4 \rfloor \lfloor 4_0 \rfloor \dots \lfloor 4_0 \rfloor \lfloor 4_1 \rfloor B a_{j_1} \dots a_{j_n} \\
\Rightarrow\ & \dots \lfloor 4 \rfloor \lfloor 4_0 \rfloor \dots \lfloor 4_0 \rfloor \lfloor 4_1 \rfloor \lfloor 4_2 \rfloor a_{j_1} \dots a_{j_n}
\end{aligned}
$$

Theorem

For every recursively enumerable language L there exists a PSC grammar $G = (V, T, P, S)$ such that G is a proper leftmost generator of its sentences preceded by their parses, G contains no more than nine nonterminals, $\mathrm{len}_{\max}(P) = 2$, and $L = \mathrm{lab}(G)^{+} \backslash L(G) \cap \mathrm{alph}(L)^{*}$.

Proof

Define the PSC grammar

$$G' = (\{S, A, B, C, 0, 1, \$_0, \$_1, \#\} \cup T \cup \mathrm{lab}(G'), T \cup \mathrm{lab}(G'), P', S),$$

where

$$\begin{aligned}\mathrm{lab}(G') = {} & (\mathrm{lab}(G) - \{\lfloor 3_0 \rfloor, \lfloor 3_1 \rfloor\}) \\ & \cup \{\lfloor 3_{01} \rfloor, \lfloor 3_{02} \rfloor, \lfloor 3_{03} \rfloor, \lfloor 3_{04} \rfloor, \lfloor 3_{11} \rfloor, \lfloor 3_{12} \rfloor, \lfloor 3_{13} \rfloor, \lfloor 3_{14} \rfloor\}\end{aligned}$$

Construction of P', Step 3

- Steps 1, 2, and 4 are the same as in the proof of Theorem 1

Step 3 $(u_{s_1} \dots u_{s_l} z_{a_{j_1}} \dots z_{a_{j_n}} \stackrel{?}{=} v_{s_1} \dots v_{s_l})$

Add

1. $\lfloor 3 \rfloor : (B, B) \rightarrow (\lfloor 3 \rfloor A, B)$
2.
   a $\lfloor 3_{01} \rfloor : (B, 0) \rightarrow (\#, \$_0)$
   b $\lfloor 3_{02} \rfloor : (A, \$_0) \rightarrow (C, \$_0)$
   c $\lfloor 3_{03} \rfloor : (C, 0) \rightarrow (\lfloor 3_{01} \rfloor \lfloor 3_{02} \rfloor \lfloor 3_{03} \rfloor, \$_0)$
   d $\lfloor 3_{04} \rfloor : (\$_0, \$_0) \rightarrow (\lfloor 3_{04} \rfloor A, B)$
3.
   a $\lfloor 3_{11} \rfloor : (B, 1) \rightarrow (\#, \$_1)$
   b $\lfloor 3_{12} \rfloor : (A, \$_1) \rightarrow (C, \$_1)$
   c $\lfloor 3_{13} \rfloor : (C, 1) \rightarrow (\lfloor 3_{11} \rfloor \lfloor 3_{12} \rfloor \lfloor 3_{13} \rfloor, \$_1)$
   d $\lfloor 3_{14} \rfloor : (\$_1, \$_1) \rightarrow (\lfloor 3_{14} \rfloor A, B)$

Construction of P', Step 3—Example

Example

$$
\begin{aligned}
& \ldots \lfloor 3 \rfloor A01 \ldots 01B01 \ldots 01a_{j_1} \ldots a_{j_n} \\
\Rightarrow\ & \ldots \lfloor 3 \rfloor \lfloor 3_0 \rfloor A1 \ldots 01\#B1 \ldots 01a_{j_1} \ldots a_{j_n}
\end{aligned}
$$

in G is simulated by

$$
\begin{aligned}
& \ldots \lfloor 3 \rfloor A01 \ldots 01B01 \ldots 01a_{j_1} \ldots a_{j_n} \\
\Rightarrow\ & \ldots \lfloor 3 \rfloor A01 \ldots 01\#\$_0 1 \ldots 01a_{j_1} \ldots a_{j_n} \\
\Rightarrow\ & \ldots \lfloor 3 \rfloor C01 \ldots 01\#\$_0 1 \ldots 01a_{j_1} \ldots a_{j_n} \\
\Rightarrow\ & \ldots \lfloor 3 \rfloor \lfloor 3_{01} \rfloor \lfloor 3_{02} \rfloor \lfloor 3_{03} \rfloor \$_0 1 \ldots 01\#\$_0 1 \ldots 01a_{j_1} \ldots a_{j_n} \\
\Rightarrow\ & \ldots \lfloor 3 \rfloor \lfloor 3_{01} \rfloor \lfloor 3_{02} \rfloor \lfloor 3_{03} \rfloor \lfloor 3_{04} \rfloor A1 \ldots 01\#B1 \ldots 01a_{j_1} \ldots a_{j_n}
\end{aligned}
$$

in G'

Conclusion

We have proved that

- for every RE language there is a PSC grammar which generates its sentences preceded by their parses
- this grammar generates its language in a leftmost way
- the total number of nonterminals and production length can be reduced

Future investigation

- which other grammars can be used for this type of generation?
- is it possible to generate sentences together with other useful information?