

Self-embedding を利用した 文脈自由言語に対する文法推論

棚次 奎介 (北九州大学)

(Keisuke Tanatsugu)

1. はじめに

アルファベット Σ 上の言語 L に対し, L のサンプルは $I(L) = \{+w; w \in L\} \cup \{-w; w \in \Sigma^* - L\}$ の有限部分集合で表せる. いま, ある言語のクラスを $\mathcal{L}$ として, 任意の $L \in \mathcal{L}$ に対して無限系列 $x_1, x_2, \dots \in I(L)$ と整数 n_0 が存在して, $n \geq n_0$ なら $L = L(f(x_1, x_2, \dots, x_n))$ が成り立つときアルゴリズム f を $\mathcal{L}$ に対する完全な文法推論アルゴリズムという. いままで正則言語等に対する興味ある推論アルゴリズムがいくつか提供されてきたが, 言語クラスが大きくなるに従って有効で完全な方法を見出すことは困難となる.

ここでは, 一般に文脈自由言語の持つ埋めこみの構造を抽出する手続きを与え, それをもとにまず線形言語に対する文法推論アルゴリズムを提示する. さらに規則の合成手法を付加することによって, 文脈自由文法を推論する完全なアルゴリズムへの拡張を行なう.

2. 定義と準備

文脈自由文法 (以下, CFG と略記する) は一般に $G = (\Gamma, \Sigma, P, S)$ で表される. ここに, Γ は非終端記号の有限集合, Σ は終端記号の有限集合で $\Gamma \cap \Sigma = \emptyset$ とする. P は $A \rightarrow x$ ($A \in \Gamma, x \in (\Gamma \cup \Sigma)^*$) なる型の生成規則の有限集合で $S \in \Gamma$ は開始記号である. $A \in \Gamma$ から生成される言語 $\{w; A \Rightarrow w, w \in \Sigma^*\}$ を L_A と書く. とくに L_S を $L(G)$ で表し, 文法 G によって生成された文脈自由言語 (以下, CFL と略記する) という. さてここでは, 文法に次のような制限を設ける:

1. 任意の $A \in \Gamma$ に対し

(1) 適当な $u, v \in (\Gamma \cup \Sigma)^*$ が存在して $S \Rightarrow uAv$ なる導出が存在する.

(2) $L_A \neq \emptyset$.

(3) $(A \rightarrow A) \notin P$.

(4) $A \neq S$ なら $A \rightarrow \alpha A \beta$ なる型の生成規則が少なくとも一つ存在する ($\alpha, \beta \in (\Gamma \cup \Sigma)^*$).

2. 任意の $A, B \in \Gamma$ に対して $L_A \neq L_B$.

以上のような制限にもかかわらず, 任意の CFL $L \subset \Sigma^*$ に対して $L = L(G)$ なる CFG G が存在する.

生成規則を $A \rightarrow uBv$ ($A, B \in \Gamma, u, v \in \Sigma^*$) または $A \rightarrow w$ ($A \in \Gamma, w \in \Sigma$) の型に限定した文法を線形文法といい, それから生成される言語を線形言語という. L を Σ 上の言語とすると, $(u, v) \in \Sigma^* \times \Sigma^*$ に対する L の導言語を

$$\bar{u}L\bar{v} = \{x; uxv \in L\}$$

と定める. このとき (u, v) を $\bar{u}L\bar{v}$ に対する L の覆いという. また, $L \subset \bar{u}L\bar{v}$ のとき L は (u, v) に関して埋めこみという.

補題 1. (1) $L_1, L_2, L \subset \Sigma^*$ のとき

$$\textcircled{1} \bar{u}^i(\bar{u}L\bar{v})\bar{v}^i = \bar{u}u^iL\bar{v}^i\bar{v}$$

$$\textcircled{2} L_1 \subset L_2 \Rightarrow \bar{u}L_1\bar{v} \subset \bar{u}L_2\bar{v}$$

$$\textcircled{3} \bar{u}(L_1 \cup L_2)\bar{v} = \bar{u}L_1\bar{v} \cup \bar{u}L_2\bar{v}, \quad \bar{u}(L_1 \cap L_2)\bar{v} = \bar{u}L_1\bar{v} \cap \bar{u}L_2\bar{v}$$

(2) CFG $G = (\Gamma, \Sigma, P, S)$ において

$$A \Rightarrow uBv \quad (A, B \in \Gamma, u, v \in \Sigma^*) \text{ なら } L_B \subset \bar{u}L_A\bar{v}.$$

Σ^* 上の辞書式順序をもとに $\Sigma^* \times \Sigma^*$ に次のような全順序関係を導入する:

$$(u_1, v_1) < (u_2, v_2)$$

$\Rightarrow$

$$(|u_1v_1| < |u_2v_2|) \vee (|u_1v_1| = |u_2v_2| \wedge u_1v_1 < u_2v_2) \vee (u_1v_1 = u_2v_2 \wedge v_1 < v_2)$$

例えば $\Sigma = \{a, b\}$ のとき $(\epsilon, \epsilon) < (a, \epsilon) < (\epsilon, a) < (b, \epsilon) < (\epsilon, b) < (aa, \epsilon) < (a, a) < (\epsilon, aa) < (ab, \epsilon) < (a, b) < \dots$ となる. また, (u, v) の次の組を $(u, v)'$ と書くことにする.

定義 1. $X, X' \subset \Sigma^*$, $(\alpha, \beta) \in \Sigma^* \times \Sigma^*$ のとき

- (1) $\mathcal{E}_X(\alpha, \beta) = \{X' \neq \emptyset; X' \subset \bar{\alpha} X' \bar{\beta}, X' \subset X\}$
- (2) $M_X(\alpha, \beta) = \bigcup \mathcal{E}_X(\alpha, \beta)$
- (3) $\mathcal{E}_X = \{M_X(\alpha, \beta); (\alpha, \beta) \in \Sigma^* \times \Sigma^*, (\alpha, \beta) \neq (\varepsilon, \varepsilon)\}$
- (4) $\hat{M}_{X'; X}(\alpha, \beta) = X' \cap M_X(\alpha, \beta)$. X が明らかな場合は単に $\hat{M}_{X'}(\alpha, \beta)$ と書く.

上の定義から直接次の事実が導かれる:

- (1) $M_X(\alpha, \beta) \subset X$. とくに $M_X(\varepsilon, \varepsilon) = X$.
- (2) $M_X(\alpha, \beta) \subset \bar{\alpha} M_X(\alpha, \beta) \bar{\beta}$.
- (3) X が有限のとき $\mathcal{E}_X = \emptyset$.

例 1. $X = \{a^m b^m; m \geq 0\} \cup \{a^n b^{2n}; n \geq 0\}$ のとき

$$\begin{aligned}\mathcal{E}_X(a, b) &= \{\{a^m b^m; m \geq 0\}, \{a^m b^m; m \geq 1\}, \dots\} \\ \mathcal{E}_X(a, b b) &= \{\{a^n b^{2n}; n \geq 0\}, \{a^n b^{2n}; n \geq 1\}, \dots\} \\ M_X(a, b) &= M_X(a^2, b^2) = \dots = \{a^m b^m; m \geq 0\} \\ M_X(a, b b) &= M_X(a^2, b^4) = \dots = \{a^n b^{2n}; n \geq 0\} \\ \mathcal{E}_X &= \{\{a^m b^m; m \geq 0\}, \{a^n b^{2n}; n \geq 0\}\}\end{aligned}$$

補題 2. CFG を $G = (\Gamma, \Sigma, P, S)$ とし, $A \in \Gamma$ とする. そのとき

$$S \Rightarrow u A v, \quad A \Rightarrow \alpha A \beta$$

ならば

$$L_A \subset M_{\bar{u} L(\hat{\alpha}) \bar{v}}(\alpha, \beta)$$

ここに $(u, v), (\alpha, \beta) \in \Sigma^* \times \Sigma^*$ である.

補題 3. CFG を $G = (\Gamma, \Sigma, P, S)$ とし, $L = L(G)$ とする.

$$S \Rightarrow u A v, \quad A \Rightarrow u' B v', \quad A \Rightarrow \alpha A \beta, \quad B \Rightarrow \alpha' B \beta'$$

のとき

$$L_B \subset \bar{u}' M_{\bar{u} L \bar{v}}(\alpha, \beta) \bar{v}' \cap M_{\bar{u} \bar{u}' L \bar{v} \bar{v}'}(\alpha', \beta')$$

ここに $(u, v), (u', v'), (\alpha, \beta), (\alpha', \beta') \in \Sigma^* \times \Sigma^*$.

いま $X \subset \Sigma^*$, $(\alpha, \beta) \in \Sigma^* \times \Sigma^*$ に対して $X_k(\alpha, \beta) = \bigcap_{i=0}^k \bar{\alpha}^i X \bar{\beta}^i$ とおき, $k \rightarrow \infty$ のとき $X_\infty(\alpha, \beta)$ とかく.

補題 4. $X \subset \bar{\alpha} X \bar{\beta}$ のとき

$$M_X(\alpha, \beta) = X_\infty(\alpha, \beta)$$

ここに $(\alpha, \beta) \in \Sigma^* \times \Sigma^*$, $X \subset \Sigma^*$.

3. 推論アルゴリズム

3.1. 非終端記号の生成

$L = L(G)$ で $G = (\Gamma, \Sigma, P, S)$ は未知とする. L のサンプル I を与えて L の文法 $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, \hat{S})$ を推論するために, Γ に対応する $\hat{\Gamma}$ を特定することから始める. いま $S \Rightarrow u A v$ かつ $(A \rightarrow \alpha A \beta) \in P$ とすれば

$$L_A \subset M_{\bar{u} L \bar{v}}(\alpha, \beta) \subset \bar{u} L \bar{v}$$

なので, $A \in P$ には上の関係を満たす適当な (u, v) を見出し, $M_{\bar{u} L \bar{v}}(\alpha, \beta)$ を対応させるのが自然である. $M_{\bar{u} L \bar{v}}(\alpha, \beta)$ は一般に無限なので実際は有限部分集合

$$\hat{M}_{\bar{u} I \bar{v}; \bar{u} L \bar{v}}(\alpha, \beta) = \bar{u} I_+ \bar{v} \cap M_{\bar{u} L \bar{v}}(\alpha, \beta)$$

を対応させ, これを G の非終端記号と考えることにする. そこでまず一般に有限集合 $X \cap W_k(\alpha, \beta)$ ($X \subset W \subset \Sigma^*$) を構成するための手続きを示す.

アルゴリズム 1. (α, β) による導言語
 入力: 有限集合 $X \subset \Sigma^*$, $(\alpha, \beta) \in \Sigma^* \times \Sigma^*$
 出力: $Y = X \cap W_k(\alpha, \beta)$
procedure DER($X, (\alpha, \beta)$):
begin
 $k := 0$; $X' := X$; $Y := \emptyset$;
 while $X' \neq Y$ **do**
 begin
 $Y := X'$;
 $k := k + 1$;
 $x' := Y \cap \overline{\alpha^k W \beta^k}$
 end;
return Y
end

アルゴリズム 1 における $Y \cap \overline{\alpha^k W \beta^k}$ なる演算は Y が有限集合なので $\overline{\alpha^k W \beta^k}$ が無限の場合でも実行可能である。ただし任意の $x \in \Sigma^*$ に対しいつでも $x \in W$ か否かが知れるものとする。さて Y はステップ毎に真に減少するので必ず停止する。

次に適当な有限集合 $X \subset W$ のもとではアルゴリズム 1 の出力は $\hat{M}_{X,W}(\alpha, \beta)$ に一致することを示そう。そのような X はある有限集合 $X_0 \subset W$ を次のように拡張して得られる:

手続き EXT

- ステップ 1. $X \leftarrow X_0$.
 ステップ 2. $(\alpha, \beta) \leftarrow (\varepsilon, \varepsilon)$.
 ステップ 3. $(\alpha, \beta) \leftarrow (\alpha, \beta)'$, $k \leftarrow 1$.
 ステップ 4. $X \cap W_{k-1}(\alpha, \beta) \subset M_W(\alpha, \beta)$ なら ステップ 7 へ.
 ステップ 5. $X \cap W_{k-1}(\alpha, \beta) \neq X \cap W_k(\alpha, \beta)$ なら $k \leftarrow k + 1$ として ステップ 4 へ.
 ステップ 6. $X \cap W_{k-1}(\alpha, \beta) - M_W(\alpha, \beta)$ の各要素 x に対して $x, \alpha x \beta, \dots, \alpha^p x \beta^p \in W$ かつ $\alpha^{p+1} x \beta^{p+1} \notin W$ なる整数 p が対応するが、それらの最小値を p_0 とし、 p_0 を実現する x の一つを x_0 とかく。 $X \leftarrow X \cup \{\alpha x_0 \beta, \dots, \alpha^{p_0-k+1} x_0 \beta^{p_0-k+1}\}$ とし、ステップ 2 へ.
 ステップ 7. $(\alpha, \beta) < (\alpha_0, \beta_0)$ なら ステップ 3 へ.
 ステップ 8. 停止.

上の手続きはいかなる有限集合 X_0 に対しても、有限ステップで停止する。さて、例えば空でない各 $M_W(\alpha, \beta)$ から一つずつ要素を選んでできる集合を X_0 とし、これを手続き EXT で拡張した集合 X は次の定理を満たす。

定理 1. $(\alpha, \beta) \in \Sigma^* \times \Sigma^*$ とする。任意の $(\alpha, \beta) \leq (\alpha_0, \beta_0)$ に対して次のことが成り立つような有限集合 $X \subset W$ が存在する:

- (1) $M_W(\alpha, \beta) = \emptyset$ なら アルゴリズム 1 の停止段階で $Y = \emptyset$.
- (2) $M_W(\alpha, \beta) \neq \emptyset$ なら アルゴリズム 1 の停止段階で $Y = \hat{M}_{X,W}(\alpha, \beta) \neq \emptyset$.

例 2. $W = \{a^m b^m; m \geq 0\} \cup \{a^n b^{2n}; n \geq 0\}$ に対して、 $X = \{\varepsilon, a b, a b b\} \subset W$ とするとアルゴリズム 1 の出力は

- (1) $(\alpha, \beta) = (a, b)$ のとき $Y = \{\varepsilon, a b\} = X \cap M_W(a, b)$
- (2) $(\alpha, \beta) = (a, b b)$ のとき $Y = \{\varepsilon, a b b\} = X \cap M_W(a, b b)$
- (3) その他のとき $Y = \emptyset$.

3.2. 生成規則の構成

適当な $X \subset W$ のもとでは アルゴリズム 1 の出力は $Y = \hat{M}_{X;W}(\alpha, \beta)$ となり得る。そこでそれらの出力の中から新しい非終端記号に相当するものを選びだし、これに関連する生成規則を構成するのが次の課題となる。

アルゴリズム 2. 線形文法の構成

入力: $(u, v), (\alpha, \beta) \in \Sigma^* \times \Sigma^*, Y, n_0 \in \mathbb{N}, r, r_0, \hat{G}$

出力: 可能な新しい変数 A と関連する生成規則の全体 $\hat{G}, r$

変数:

$\mu(i)$: i 番目の変数 $A_i \in \hat{\Gamma}$ に対応する有限集合

$\hat{\Gamma}(u, v) \subset \hat{\Gamma}$: (u, v) のもとで生みだされる変数の集合

$SUB \subset \Sigma^* \times \Sigma^* \times \mathbb{N}$: 非埋めこみの規則を構成させる。 $((u', v'), x) \in SUB$ なら $A_x \rightarrow u' A_r v'$ が対応。

$SUB1 \subset \mathbb{N}$: $x \in SUB1$ なら $A_x \rightarrow A_r$ が対応。

$SUB2 \subset \mathbb{N}$: $x \in SUB2$ なら $A_r \rightarrow A_x$ が対応。

r_0 : (u, v) のもとで最初に生み出される変数の番号。

check: Y に対して新しい変数を考えることができないなら 0, (u, v) のもとではじめての変数が対応するなら 1, 2 度目以上なら 2 の値をもつ。

procedure LG($Y, (u, v), (\alpha, \beta), n_0, r, r_0, \hat{G}$):

begin

check := 0;

if $Y \neq \emptyset$ then

begin

if $r < r_0$ then check := 1

else if $Y \neq \mu(i)$ for all i ($r_0 \leq i \leq r$) then check := 2;

if check $\neq$ 0 then

begin

SUB := $\emptyset$; SUB1 := $\emptyset$; SUB2 := $\emptyset$;

if $r \neq 1$ then

for $i = 1$ until $r_0 - 1$ do

if $Y \cap \overline{\mu(i)} \neq \emptyset$ for some (u', v') such that

$(u, v) = (u'' u', v' v'')$ and $A_i \in \hat{\Gamma}(u'', v'')$ then

SUB := SUB $\cup \{ (u', v'), i \}$;

if check = 2 then

for $i = r_0$ until r do

if $Y \subset \mu(i)$ then SUB1 := SUB1 $\cup \{ i \}$

else if $Y \supset \mu(i)$ then SUB2 := SUB2 $\cup \{ i \}$;

$r := r + 1$;

$\mu(r) := Y$; $\hat{\Gamma}(u, v) := \hat{\Gamma}(u, v) \cup \{ A_r \}$; $\hat{\Gamma} := \hat{\Gamma} \cup \{ A_r \}$;

$\hat{P} := \hat{P} \cup \{ A_r \rightarrow \alpha A_r \beta \} \cup \{ A_x \rightarrow u' A_r v' ; ((u', v'), x) \in SUB \}$

$\cup \{ A_x \rightarrow A_r ; x \in SUB1 \} \cup \{ A_r \rightarrow A_x ; x \in SUB2 \}$

$\cup \{ A_r \rightarrow w ; |w| \leq n_0, w \in \mu(r) \}$

end

else if there is no derivation such that $A_i \Rightarrow \alpha A_i \beta$ then

$\hat{P} := \hat{P} \cup \{ A_i \rightarrow \alpha A_i \beta \}$

end;

return $\hat{G}$

end

3.3. 線形文法の推論

ここでは、言語 L の正サンプル $I_+ \subset L$ に対し、非終端記号特定の際の覆いに関する探索上限を (u_0, v_0) 、埋めこみ性に関する探索上限を (α_0, β_0) および右辺が終端記号列からなる生成規則を確定するための記号列の長さの上限 n_0 を与えて線形文法 $\hat{G}$ を構成するための手続きを示す。

アルゴリズム3. 線形文法の推論

入力: 有限集合 $I_+ \subset \Sigma^*$, (u_0, v_0) , $(\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, $n_0 \in \mathbb{N}$

出力: 線形文法 $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, A_1)$

procedure LG_INF (I_+ , (u_0, v_0) , (α_0, β_0) , n_0):

begin

$\hat{\Gamma} := \{A_1\}$; $\hat{\Gamma}(\varepsilon, \varepsilon) := \{A_1\}$; $\hat{P} := \{A_1 \rightarrow w; |w| \leq n_0, w \in I_+\}$; $(u, v) := (\varepsilon, \varepsilon)$;

$r := 1$; $r_0 := 1$; $\mu(1) := I_+$;

while $(u, v) \leq (u_0, v_0)$ **do**

if $I_+ \not\subset L(\hat{G})$ **then**

begin

$X := \bar{u} I_+ \bar{v}$;

if $X \neq \emptyset$ **then**

begin /* (u, v) のもとでの規則の構成 */

$(\alpha, \beta) := (\varepsilon, \varepsilon)'$;

while $(\alpha, \beta) \leq (\alpha_0, \beta_0)$ **do**

begin

$Y := \text{DER}(X, (\alpha, \beta))$;

$\hat{G} := \text{LG}(Y, (u, v), (\alpha, \beta), n_0, r, r_0, \hat{G})$;

$(\alpha, \beta) := (\alpha, \beta)'$

end

end;

$(u, v) := (u, v)'$;

$r_0 := r + 1$

end

else $(u, v) := (u_0, v_0)'$; /* $I_+ \subset L(\hat{G})$ のとき終了 */

return $\hat{G}$

end

例3. $L = \{a b^m a b^n a; m, n \geq 0\}$, $I_+ = \{a a a, a b a b a\}$, $(u_0, v_0) = (a, a)$, $(\alpha_0, \beta_0) = (\varepsilon, b)$, $n_0 = 1$ とするとき線形文法 $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, A_1)$ が表1のように推論される。

4. 線形言語に対する完全性

ここでは任意の線形言語 L に対し、適当なサンプル $I_+ \subset L$, (u_0, v_0) , $(\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, 整数 n_0 を与えるとアルゴリズム3は $L = L(\hat{G})$ なる線形文法を推論できることを示す。

補題5. 任意の言語 L に対し、次のような条件を満たす線形文法 $G = (\Gamma, \Sigma, P, [L])$ を考えると $L(G) \subset L$ である:

- (1) Γ は $\{[X]; X \subset \bar{u} L \bar{v} \text{ なる } (u, v) \in \Sigma^* \times \Sigma^* \text{ が存在し, } X \neq \emptyset\}$ のある有限部分集合に $[L]$ を加えた集合。
- (2) $([X_1] \rightarrow u [X_2] v) \in P$ ならば $X_2 \subset \bar{u} X_1 \bar{v}$, $([X] \rightarrow w) \in P$ ならば $w \in X$.

補題6. アルゴリズム3において、 $I_+ \subset L$ に対し $M_L(\alpha, \beta) \in \mathcal{E}_{\bar{u} L \bar{v}}$, $\bar{u} I_+ \bar{v} \neq \emptyset$ ($(u, v) \leq (u_0, v_0)$, $(\alpha, \beta) \leq (\alpha_0, \beta_0)$) のとき適当な整数 m が存在して $\mu(m) = \hat{M}_{\bar{u} I_+ \bar{v}}(\alpha, \beta)$.

補題 7. $G=(\Gamma, \Sigma, P, S)$ を線形文法とし, $L=L(G)$ とする. そのとき適当な $I_+ \subset L$, $(u_0, v_0), (\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, 整数 n_0 のもとでアルゴリズム 3 は次のような文法 $\hat{G}=(\hat{\Gamma}, \Sigma, \hat{P}, A_1)$ を推論する: 任意の $A \in \Gamma$ に対し, $\hat{A} \in \hat{\Gamma}$ が存在して $L_A \subset L_{\hat{A}}$.

[略証] $(u_0, v_0) = \max \{ (u, v) ; S \Rightarrow x \Rightarrow u y v, x \text{ の導出には各 } A \in \Gamma \text{ が高々一度現れる, } x, y \in (\Gamma \cup \Sigma)^* \}$

$(\alpha_0, \beta_0) = \max \{ (\alpha, \beta) ; (A \rightarrow \alpha A \beta) \in P, A \in \Gamma \}$

$n_0 = \max \{ |w| ; (A \rightarrow w) \in P, w \in \Sigma^*, A \in \Gamma \}$

$I_+ = \{ x ; x \leq u_0 \alpha_0 \beta_0 v_0, x \in L \}$

とすると, 任意の $A \in \Gamma$ に対して $S \Rightarrow u A v$ ($(u, v) \leq (u_0, v_0)$) なる導出が存在する. さらに A からのすべてのルールを追跡できるので $\hat{G}$ における生成規則の集合 $\hat{P}$ は P に対応するものを含んでいることが理解できる. また, A, B , に対応して $\hat{A}, \hat{B} \in \hat{\Gamma}$ が得られたとすれば

$$A \neq B \Leftrightarrow \hat{A} \neq \hat{B}$$

以上から $L_A \subset L_{\hat{A}}$.

定理 2. 任意の線形言語 L に対し, 適当なサンプル $I_+ \subset L$, $(u_0, v_0), (\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, 整数 n_0 が存在して, アルゴリズム 3 は $L=L(\hat{G})$ なる線形文法 $\hat{G}$ を推論する.

[略証] 補題 7 で $A=S$ の場合を考えると $L \subset L(\hat{G})$. またアルゴリズム 3 において, $\mu(r) = \hat{M}_{\Gamma I_+ \Gamma}(\alpha, \beta)$ のとき A_r は $[M_{\Gamma I_+ \Gamma}(\alpha, \beta)]$ を表すと解釈すれば, 推論される文法 $\hat{G}$ は補題 5 の条件を満足するので $L(\hat{G}) \subset L$ となる. よって $L=L(\hat{G})$ を得る.

(u, v)	(α, β)	Y	$\hat{\Gamma}$	$\hat{P}$	$L(\hat{G})$
$(\varepsilon, \varepsilon)$	$(\varepsilon, \varepsilon)$	I_+	A_1	ϕ	ϕ
$(\varepsilon, \varepsilon)$	(a, ε) (ε, a) (b, ε) (ε, b)	ϕ ϕ ϕ ϕ		ϕ	ϕ
(a, ε)	(a, ε) (ε, a) (b, ε) (ε, b)	ϕ ϕ $\{a a, b a b a\}$ ϕ	A_2	$A_1 \rightarrow a A_2, A_2 \rightarrow b A_2$	ϕ
(ε, a)	(a, ε) (ε, a) (b, ε) (ε, b)	ϕ ϕ ϕ $\{a a, a b a b\}$	A_3	$A_1 \rightarrow A_3 a, A_3 \rightarrow A_3 b$	ϕ
$\cdot$ $\cdot$ $\cdot$	$\cdot$ $\cdot$ $\cdot$				
(a, a)	(a, ε) (ε, a) (b, ε) (ε, b)	ϕ ϕ $\{a, b a b\}$ $\{a, b a b\}$	A_4	$A_1 \rightarrow a A_4 a, A_4 \rightarrow b A_4, A_4 \rightarrow a A_4 \rightarrow A_4 b$	L

表 1. $I_+ = \{a a a, a b a b a\}$ のときの アルゴリズム 3 の推論過程

5. 文脈自由言語に対する文法推論

CFL L に対し, サンプルとして $I_+ \subset L$ のみならず負サンプル $I_- \subset \Sigma^* - L$ をも追加して, $L = L(\hat{G})$ なる CFG $\hat{G}$ を同定するための手続きを考える. そのために, アルゴリズム 3 で得られる線形文法を合成して, 生成規則の右辺に二つ以上の非終端記号を含み得るようにする.

アルゴリズム 4. 文形式の合成

入力: $x, y \in (\hat{\Gamma} \cup \Sigma)^*$, $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, A_1)$

出力: x と y の合成語全体 H

変数 m : 現在解析中の語の番号

n : 合成途中の語の最大番号

$z(m)$: 現時点までの合成語

procedure COMPOUND(x, y):

begin

$m := 1$; $n := 1$; $z(1) := \varepsilon$; $x(1) := x$; $y(1) := y$; $H := \emptyset$;

while $m \leq n$ do

begin

delete the maximal length common prefix of $x(m)$ and $y(m)$;

if $x(m) = Ax'$ for some $A \in \hat{\Gamma}$ and $x' \in (\hat{\Gamma} \cup \Sigma)^*$, and there exist $w \in \Sigma^*$ and $y' \in (\hat{\Gamma} \cup \Sigma)^*$ such that $A \Rightarrow w$ and $y(m) = wy'$ then

begin

let $A \Rightarrow w_1, \dots, A \Rightarrow w_l$ and $y(m) = w_1 y'_1 = \dots = w_l y'_l$;

for $j = n+1$ until $n+l$ do

begin

$x(j) := x'$; $y(j) := y'_j$; $z(j) := z(m)A$

end;

$n := n+l$

end

else

if $y(m) = By'$ for some $B \in \hat{\Gamma}$ and $y' \in (\hat{\Gamma} \cup \Sigma)^*$ then

if there exist $w \in \Sigma^*$ and $x' \in (\hat{\Gamma} \cup \Sigma)^*$ such that $A \Rightarrow w$ and $x(m) = wx'$ then begin

let $B \Rightarrow w_1, \dots, B \Rightarrow w_l$ and $x(m) = w_1 x'_1 = \dots = w_l x'_l$;

for $j = n+1$ until $n+l$ do

begin

$x(j) := x'_j$; $y(j) := y'$; $z(j) := z(m)B$

end

end

else if $x(m) = y(m) = \varepsilon$ then $H := H \cup \{z(m)\}$;

$m := m+1$

end;

return H

end

アルゴリズム 5. 文脈自由文法の推論

入力: 有限集合 $I_+ \subset \Sigma^*$, (u_0, v_0) , $(\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, $n_0 \in \mathbb{N}$

出力: 文脈自由文法 $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, A_1)$

変数 check: すべての変数に対して合成不可能となったとき 0, その他のとき 1

$c(i)$: A_i からの生成規則が一つも合成されないとき 0, その他のとき 1

```

procedure CFG_INF( $I_+$ , ( $u_0$ ,  $v_0$ ), ( $\alpha_0$ ,  $\beta_0$ ),  $n_0$ ) :
begin
  check:=1;
   $\hat{G} := LG\_INF(I_+, (u_0, v_0), (\alpha_0, \beta_0), n_0)$ ;
  for all  $i$  ( $1 \leq i \leq r$ ) do  $c(i) := 1$ ;
  while  $I_+ \not\subseteq L(\hat{G})$  do
    if check  $\neq 0$  then
      begin
        check:=0;
        for  $i=1$  until  $r$  do
          if  $c(i)=1$  then
            begin
              check:=1;  $c(i) := 0$ ;
              let  $n(i)$  be the number of production rules from  $A$  and  $x_1, \dots, x_{n(i)}$ 
                be right-side strings of these rules;
              for  $p=1$  until  $n(i)$  do
                for  $q=p$  until  $n(i)$  do
                  if COMPOUND( $x_p, x_q$ )  $\neq \emptyset$  then
                    begin
                      let COMPOUND( $x_p, x_q$ ) =  $\{y_1, \dots, y_t\}$ ;
                      for  $j=1$  until  $t$  do
                        begin
                           $\hat{P} := \hat{P} \cup \{A_i \rightarrow y_j\}$ ;
                          if  $I_- \cap L(\hat{G}) \neq \emptyset$  then  $\hat{P} := \hat{P} - \{A_i \rightarrow y_j\}$ 
                        end
                      else  $c(i) := 1$ 
                    end
                  end
                end
              end
            end
          end
        end
      else return  $\emptyset$ ;
  return  $\hat{G}$  /*  $\hat{G}$  is compatible to given sample ( $I_+$ ,  $I_-$ ) */
end

```

定理 3. 任意の CFL L に対し、適当なサンプル (I_+, I_-) , (u_0, v_0) , $(\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$, 整数 n_0 が存在して、アルゴリズム 5 は $L = L(\hat{G})$ なる CFG $\hat{G} = (\hat{\Gamma}, \Sigma, \hat{P}, A_1)$ を推論する。

[略証] $L = L(G)$ かつ $G = (\Gamma, \Sigma, P, S)$ としよう。 G の生成規則が $A \rightarrow u_0 A_1 u_1 \cdots A_m u_m$ の場合、定理 2 と同様の論法により適当な I_+ , (u_0, v_0) , $(\alpha_0, \beta_0) \in \Sigma^* \times \Sigma^*$ および整数 n_0 が存在して、 $(\hat{A} \rightarrow u_0 w, u_1 \cdots u_{i-1} \hat{A}_i u_i \cdots w_m u_m) \in \hat{P}$ ($i=1, 2, \dots, m$) となる。ただし W_j は L_{A_j} の最小長語の一つとする ($j=1, 2, \dots, m$)。規則の合成法により $(\hat{A} \rightarrow u_0 \hat{A}_1 u_1 \cdots \hat{A}_m u_m) \in \hat{P}$ が得られる。誤った合成については I_- を充分大きくすることによってチェック可能である。

例 4. $L = \{a b^m a^m b a^n b^n a; m, n \geq 0\}$, $I_+ = \{a b a, a b a b a, a b a b a b a\} \subset L$, $I_- = \{a b a a b b a b a\} \not\subset L$, $(u_0, v_0) = (a, b a)$, $(\alpha_0, \beta_0) = (b, a)$, $n_0 = 0$ のもとにアルゴリズム 5 を適用するとまず、関数手続き LG_INF によって表 2 に示すように線形文法 $\hat{G} = (\{A_1, A_2, A_3\}, \{a, b\}, \hat{P}, A_1)$ が得られる。ただし、

$\hat{P} = \{A_1 \rightarrow a b A_2 a / a A_3 b a, A_2 \rightarrow a A_2 b / \varepsilon, A_3 \rightarrow b A_3 a / \varepsilon\}$
 である。 $A_1 \in \hat{\Gamma}$ から順に生成規則の合成を行なうと

(1) COMPOUND($a b A_2 a, a b A_2 a$) = $\{a b A_2 A_2 a\}$:

$(A_1 \rightarrow a b A_2 A_2 a)$ を $\hat{P}$ につけ加えると $a b a a b b a b a \notin I_+$ を生成して矛盾。

(2) COMPOUND $(a b A_2 a, a A_3 b a) = \{a A_3 b A_2 a\}$:

$(A \rightarrow a A_3 b A_2 a)$ を $\hat{P}$ に付加すると $I_+ \subset L(\hat{G})$ かつ $I_- \cap L(\hat{G}) = \emptyset$ となって終了。
 実際, $L = L(\hat{G})$ となる.

(u, v)	(α, β)	Y	$\hat{F}$	$\hat{P}$	$L(\hat{G})$
$(\varepsilon, \varepsilon)$	$(\varepsilon, \varepsilon)$	I_+	A_1	$\emptyset$	$\emptyset$
$\cdot$ $\cdot$ $\cdot$	$\cdot$ $\cdot$ $\cdot$				
$(a b, a)$	$\cdot$ (a, b) $\cdot$ $\cdot$	$\{\varepsilon, a b\}$	A_2	$A_1 \rightarrow a b A_2 a, A_2 \rightarrow a A_2 b, A_2 \rightarrow \varepsilon$	$abababa \notin L(\hat{G})$
$(a, b a)$	$\cdot$ (b, a)	$\{\varepsilon, b a\}$	A_3	$A_1 \rightarrow a A_3 b a, A_3 \rightarrow b A_3 a, A_3 \rightarrow \varepsilon$	$abababa \notin L(\hat{G})$

表2. $I_+ = \{a b a, a b a b a, a b a b a b a\}$ のときの アルゴリズム5 の推論過程

REFERENCES

- (1) R.Soromonoff, A Formal Theory of Inductive Inference, Information and Control, Vol.7, pp.1-22, pp.224-254, 1964.
- (2) S.Huzino, On Some Properties of Derivative-mappings, Structural Diagrams and Structural Equations: Part1, Memo.Fac.Sci.Kyusyu Univ. Ser.A20, pp179-265, 1966.
- (3) M.Gold, Language Identification in the Limit, Information and Control, Vol.10, pp447-474, 1967.
- (4) J.A.Feldman, J.Gips, J.J.Horning and S.Redner, "Grammatical Complexity and Inference", Technical Report No.CS125, Computer Science Department, Stanford University, 1969.
- (5) A.W.Biermann, An Interactive Finite-state Language Learner, Proc. 1-st USA-JAPAN Comp.Conf. (1972), 13-20.
- (6) K.Tanatsugu and S.Arikawa, On Characteristic Sets and Degrees of Finite Automata, International Journal of Computer and Information Sciences, Vol.6, No.1, 1977.
- (7) K.Tanatsugu, A Grammatical Inference for Harmonic Linear Languages, International Journal of Computer and Information Sciences, Vol.13, No.5, 1984.