条件付確率の変更による確率一般化LR構文解析法の拡張

椎名 広光: 岡山理科大学 総合情報学部
増山 繁: 豊橋技術科学大学 知識情報工学系

1 はじめに

確率構文解析法は，音声認識における音素の予測モデル [6] などの統計的言語モデルとして用いられてきている．これは文脈自由文法の生成規則に出現確率を付加したモデルで, inside-outside アルゴリズムなどによって生成規則を推定し，尤もらしい構文解析木を求めるモデルが知られている．また，プッシュダウンオートマトンを予め作成しておいて，それを用いて構文解析を進めるLR構文解析法を，文脈自由言語に対する構文解析法に拡張した一般化LR構文解析法 [1](以降，GLR構文解析) が知られている．

Briscoe と Carroll は，GLR構文解析法を応用して状態と先読み文字列の組合せに対して出現確率を付加する確率GLR構文解析法 [2](以降，B&C法と呼ぶ) を提案している．しかし，出現確率の正規化に問題があり，構文解析木の正しい生起確率を求めることができない場合があった．それに対し，Inui ら [2] によって提案されている確率GLR構文解析 (以降，Inui 法と呼ぶ) では，出現確率の情報を持つ状態と先読み文字列の組合せの方法を工夫することによって，B&C 法で起こる出現確率の正規化の問題を解決し，構文解析木の生起確率を厳密なものとしている．

しかしながら，学習させる構文解析木が複雑な場合は，B&C 法や Inui 法においては，現在の状態と先読み文字列及び次の予測される状態の組合せだけでは，構文解析木の正しい生起確率を計算することができない．そこで，本稿では学習させる構文解析木を用いて出現確率を，次の入力文字，これから遷移する先読み状態 (以降，先読み状態)，解析中にスタックに積まれている状態ごとに求めるように拡張する．そして解析途中のスタックに積まれている状態の個数と先読み状態の個数が十分な個数であれば，拡張した確率一般化LR構文解析により構文解析木の正しい生起確率を求めることができることを示す．

2 準備

与えられた文脈自由文法 $G = (P, N, T, S, \$)$, P: 生成規則の集合, N: 非終端記号の集合, T: 終端記号, S: 開始記号，$\$$: 入力の終りを示す特殊記号から作られるLR状態遷移図 $\mathbf{LRA} = (\mathbf{S}, \delta)$ は次のように定義される．

(1) $\mathbf{S}$: LR状態遷移図の状態集合．状態はプッシュダウンオートマトンの動作をあらわすため，状態集合は **shift** 状態集合と **reduce** 状態集合と受理を表す s_{acc} に分けられ，$\mathbf{S} = \mathbf{S_S} \cup \mathbf{S_R} \cup \{s_{acc}\}$ となる．また，開始状態を s_0 と表し，$s_0 \in \mathbf{S_S}$ である．

(2) δ: LR状態遷移図の各状態における動作を表す．状態は **shift** 状態と **reduce** 状態に分けられる．**shift** 状態では，**shift** 状態と文法記号の組合せに対して次に遷移する先の状態を関数 $\delta_{\mathbf{S}}$ で表す．

$$\delta_{\mathbf{S}}(s_i, l_i) = \{s_j \mid s_i, s_j \in \mathbf{S_S}, l_i \in T \cup N\}.$$

一方，**reduce** 状態では，**reduce** 状態で還元する文法規則の種類を関数 $\delta_{\mathbf{R}}$ で表す．

$$\delta_{\mathbf{R}}(s_i) = \{A \to \alpha \mid s_i \in \mathbf{S_R}, \text{“}A \to \alpha\text{”} \in P\}.$$

3 一般化LR構文解析法

本節では，一般化LR構文解析法のうち Inui 法について述べる．

3.1 Inui 法

Inui 法では，入力文字列 $l_1 l_2 \ldots l_n$ による開始状態 s_0 から受理状態 s_{acc} までの遷移の過程における状態を1つの状態だけでなく状態を積んでいるスタックと考えている．$\sigma_i \in \sigma$ を時点 i におけるスタックの内容とすると，遷移過程は次のように示される．

$$\sigma_0 \overset{l_1 a_1}{\Longrightarrow} \sigma_1 \overset{l_2 a_2}{\Longrightarrow} \ldots \overset{l_{n-1} a_{n-1}}{\Longrightarrow} \sigma_{n-1} \overset{l_n a_n}{\Longrightarrow} \sigma_n$$

これに対して，構文解析木 T の生起確率 $P(T)$ も同様にスタックを用いて定義している．また，時点 i における入力文字 l_i，動作 a_i，スタック σ_i は時点 $i-1$ におけるスタック σ_{i-1} に依存すると

仮定すると，次のように定義している．

$$P(T) = P(\sigma_0, l_1, a_1, \sigma_1, \ldots, l_n, a_n, \sigma_n)$$
$$= \prod_{i=1}^{n} P(l_i, a_i, \sigma_i \mid \sigma_0, l_1, a_1, \sigma_1, \ldots, l_{i-1}, a_{i-1}, \sigma_{i-1})$$
$$\approx \prod_{i=1}^{n} P(l_i, a_i, \sigma_i \mid \sigma_{i-1})$$

$P(l_i, a_i, \sigma_i \mid \sigma_{i-1})$ では，全てのスタックを保持しておかなければならないので，スタック σ_i の先頭の状態 s_i で近似している．

また，時点 i における1つ前のスタック先頭の状態 s_i が **shift** 状態または **reduce** 状態によって条件付き確率を変更している．この変更が B&C 法の方法で起こる出現確率の正規化の問題を Inui 法が解決している点である．

$$P(l_i, a_i, \sigma_i \mid \sigma_{i-1}) \approx \begin{cases} P(l_i, a_i \mid s_i), & \text{if } a_i = \textbf{shift} \\ P(a_i \mid s_{i-1}, l_i), & \text{if } a_i = \textbf{reduce} \end{cases}$$

4 拡張 PGLR

本論文で提案する方法では，構文解析時のスタックの状態と先読みとして状態遷移を用いる条件付き確率で，構文解析木 $P(T)$ の生起確率を近似する．

$$P(T) = P(\sigma_0, l_1, a_1, \sigma_1, \ldots, l_n, a_n, \sigma_n)$$
$$= \prod_{i=1}^{n-1} P(\sigma_i, l_i, a_i, \sigma_{i+1}, \ldots, l_n, a_n, \sigma_n \mid \sigma_0, l_1, a_1, \sigma_1, \ldots, l_{i-1}, a_{i-1}, \sigma_{i-1})$$
$$= \prod_{i=1}^{n-1} P(l_i, a_i, \sigma_i \sigma_{i+1} \ldots \sigma_n \mid \sigma_{i-1})$$

$\sigma_i \sigma_{i+1} \ldots \sigma_n$ については，それぞれのスタックの先頭の状態だけを取り出した $s_{1,i}, s_{1,i+1}, \ldots, s_{1,n}$ で代表させる．また，$\sigma_{i-1} = s_{1,i-1} s_{2,i-1} \cdots s_{i-1,i-1}$ であるとすると，スタックの上から k 個を表し，$\sigma_{i-1,k} = s_{1,i-1} s_{2,i-1} \cdots s_{k,i-1}$ とする．以下に条件付き確率の近似を示す，

$$P(l_i, a_i, \sigma_i \sigma_{i+1} \ldots \sigma_n \mid \sigma_{i-1})$$
$$\approx P(l_i, a_i, \sigma_i \sigma_{i+1} \ldots \sigma_{i+j-1} \mid \sigma_{i-1,k})$$
$$\approx \begin{cases} P(l_i, a_i, s_{1,i} s_{1,i+1} \cdots s_{1,i+j-1} \mid s_{1,i-1} s_{2,i-1} \cdots s_{k,i-1}), & \text{if } a_i = \textbf{shift} \\ 1, & \text{if } a_i = \textbf{reduce} \end{cases}$$
$$\approx P(l_i, s_{1,i} s_{1,i+1} \cdots s_{1,i+j-1} \mid s_{1,i-1} s_{2,i-1} \cdots s_{k,i-1})$$

これを PGLR(j,k) 構文解析法と呼び，j は先読みをする状態数を，また k は現時点でのスタックの状態数を表すものとする．j と k が十分大きければ $P(l_i, a_i, \sigma_i \sigma_{i+1} \ldots \sigma_n \mid \sigma_{i-1})$ と $P(l_i, a_i, s_{1,i} s_{1,i+1} \cdots s_{1,i+j-1} \mid s_{1,i-1} s_{2,i-1} \cdots s_{k,i-1})$ とが等しくなる．

この PGLR(j,k) 構文解析法は，正確な構文解析木を学習するためにスタックの先頭からの状態の個数 j とこれから到来する先読みとして状態の個数 k が必要となることを示している．この j と k は学習する構文解析木と文法から求めることができ，j,k が十分大きくない場合は，構文解析木の生起確率を求めることができるが正確な値は求められない．しかし，j と k が十分に大きな値であるならば，そのような構文解析法でもある程度の正確さをもった構文解析木の生起確率を求めることができると考えられる．

4.1 拡張 PGLR の例

文法 $G_1 = (P_1, N_1, T_1, S, \$)$, P_1={$S \to uS$, $S \to vS$, $S \to SS$, $S \to x$ }, N_1 = { S }, T_1 = { u, v, x } を用いて示す．ここで，文法 G_1 に対する LR 表は表1のように作成される．

表1: 文法 G_1 の LR 表

state	action				goto
	u	v	x	\$	S
S_0	**sh1**	**sh2**	**sh3**		S_4
S_1	**sh1**	**sh2**	**sh3**		S_5
S_2	**sh1**	**sh2**	**sh3**		S_6
S_3	**re3**	**re3**	**re3**		
S_4	**re3**	**re3**	**re3**	**acc**	S_7
S_5	**sh1/re1**	**sh2/re1**	**sh3/re1**	**re1**	S_7
S_6	**sh1/re2**	**sh2/re1**	**sh3/re1**	**re2**	S_7
S_7	**sh1/re4**	**sh2/re4**	**sh3/re4**	**re4**	S_7

本節では，PGLR(j,k) 構文解析法の適用例として入力データ *uuxx*, *uvxx*, *vuxx*, *vvxx* を取り上げる．

入力データごとに3種類の構文解析木があり，*uuxx* には図1に示す $T(A_{uu})$, $T(B_{uu})$, $T(C_{uu})$ の3種類がある．

構文解析木を状態遷移のリストで表すことにすると，*uuxx* に対する $T(A_{uu})$, $T(B_{uu})$, $T(C_{uu})$, *uvxx* に対する $T(A_{uv})$, $T(B_{uv})$, $T(C_{uv})$, *vuxx* に対する $T(A_{vu})$, $T(B_{vu})$, $T(C_{vu})$, *vvxx* に対する $T(A_{vv})$, $T(B_{vv})$, $T(C_{vv})$ の12個のリストがある．以下に状態遷移リストを示す．

$T(A_{uu})$: $S_0 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{x,\textbf{sh}}{\Longrightarrow} S_3 \overset{x,\textbf{re}}{\Longrightarrow} S_5 \overset{x,\textbf{re}}{\Longrightarrow} S_5 \overset{x,\textbf{sh}}{\Longrightarrow} S_3 \overset{\$,\textbf{re}}{\Longrightarrow} S_7 \overset{\$,\textbf{re}}{\Longrightarrow} S_5 \overset{\$,\textbf{re}}{\Longrightarrow} S_{acc}$

$T(B_{uu})$: $S_0 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{x,\textbf{sh}}{\Longrightarrow} S_3 \overset{x,\textbf{re}}{\Longrightarrow} S_5 \overset{x,\textbf{sh}}{\Longrightarrow} S_3 \overset{x,\textbf{re}}{\Longrightarrow} S_7 \overset{\$,\textbf{re}}{\Longrightarrow} S_5 \overset{\$,\textbf{re}}{\Longrightarrow} S_5 \overset{\$,\textbf{re}}{\Longrightarrow} S_{acc}$

$T(C_{uu})$: $S_0 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{u,\textbf{sh}}{\Longrightarrow} S_1 \overset{x,\textbf{sh}}{\Longrightarrow} S_3 \overset{x,\textbf{re}}{\Longrightarrow} S_5 \overset{x,\textbf{re}}{\Longrightarrow} S_5 \overset{x,\textbf{re}}{\Longrightarrow} S_4 \overset{\$,\textbf{sh}}{\Longrightarrow} S_3 \overset{\$,\textbf{re}}{\Longrightarrow} S_7 \overset{\$,\textbf{re}}{\Longrightarrow} S_{acc}$

$T(A_{uv})$: $S_0 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_5 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_5 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(B_{uv})$: $S_0 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{sh}}{\Longrightarrow}$
$S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_5 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(C_{uv})$: $S_0 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_5 \overset{x,\mathbf{re}}{\Longrightarrow} S_4 \overset{\$,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(A_{vu})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_5 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_6 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(B_{vu})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_5 \overset{x,\mathbf{sh}}{\Longrightarrow}$
$S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_5 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(C_{vu})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{u,\mathbf{sh}}{\Longrightarrow} S_1 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_5 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_6 \overset{x,\mathbf{re}}{\Longrightarrow} S_4 \overset{\$,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(A_{vv})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_6 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(B_{vv})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{sh}}{\Longrightarrow}$
$S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_6 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

$T(C_{vv})$: $S_0 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{v,\mathbf{sh}}{\Longrightarrow} S_2 \overset{x,\mathbf{sh}}{\Longrightarrow} S_3 \overset{x,\mathbf{re}}{\Longrightarrow} S_6 \overset{x,\mathbf{re}}{\Longrightarrow}$
$S_6 \overset{x,\mathbf{re}}{\Longrightarrow} S_4 \overset{\$,\mathbf{sh}}{\Longrightarrow} S_3 \overset{\$,\mathbf{re}}{\Longrightarrow} S_7 \overset{\$,\mathbf{re}}{\Longrightarrow} S_{acc}$

出現確率の先読み状態は，終端記号によって状態遷移を行ない，次の終端記号の処理に移るまでの処理と考えてよい．構文解析木 $T(A_{uu})$ から得られる出現確率には，次の場合がある．

最初の文字 u: $P(u, S_1 \mid S_0)$,
2番目の文字 u: $P(u, S_1 \mid S_0S_1)$,
3番目の文字 x: $P(x, S_3S_5S_5 \mid S_0S_1S_1)$,
4番目の文字 x: $P(x, S_3S_7S_5S_{acc} \mid S_0S_1S_5)$

表2は，入力データ全てに対する出現確率である．表2から分かるとおり，学習したデータから構文解析木を1個に決めるには，時点 i におけるスタックの先頭から4個の状態と先読みの5個の状態が必要である．よって，この学習例はPGLR(4, 5) 構文解析といえる．

なお，表2の中では，条件つき確率の条件が等しいもののうち，先読み状態が部分的に等しいものがある．この場合は構文解析時における出現確率のあてはめは，先読み状態の数が多い出現確率を優先することにする．つまり，(1)$P(x,S_3S_5S_5|S_0S_1S_1)$, (2)$P(x,S_3S_5\ |S_0S_1S_1)$, (3)$P(x,S_3S_5S_5S_4|\ S_0S_1S_1)$ の場合では，(1) → (3) → (2) の順であてはめる．これは，先読み文字列の数を減らすためである．

また，表2では，

$N = A_{uu} + B_{uu} + C_{uu} + A_{uv} + B_{uv} + C_{uv}$
$\quad +A_{vu} + B_{vu} + C_{vu} + A_{vv} + B_{vv} + C_{vv}$
$N_{uu} = A_{uu} + B_{uu} + C_{uu}$
$N_{uv} = A_{uv} + B_{uv} + C_{uv}$

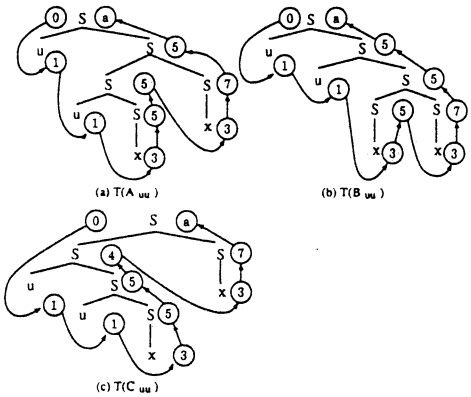

図1: 入力 $uuxx$ の構文解析木

$N_{vu} = A_{vu} + B_{vu} + C_{vu}$
$N_{vv} = A_{vv} + B_{vv} + C_{vv}$
としている．

構文解析木 $T(A_{uu})$ の生起確率 $P(T(A_{uu}))$ は，次のように計算される．

$$\begin{aligned} & P(T(A_{uu})) \\ &= P(u, S_1 \mid S_0) \times P(u, S_1 \mid S_0S_1) \\ &\quad \times P(x, S_3S_5S_5 \mid S_0S_1S_1) \\ &\quad \times P(x, S_3S_7S_5S_{acc} \mid S_0S_1S_5) \\ &= \frac{N_{uu}+N_{uv}}{N} \times \frac{N_{uu}}{N_{uu}+N_{uv}} \times \frac{A_{uu}}{N_{uu}} \times 1 \\ &= \frac{A_{uu}}{N} \end{aligned}$$

上記から，スタックの内容といくつかの状態の先読みによって構文解析木の生起確率を正しく計算できることが分かる．

4.2 PGLR(j,k) 構文解析の j,k の削減

4.1 節では PGLR(4, 5) 構文解析の例を示した．しかし，**4.1** 節の例は条件付き確率のパラメータを学習した構文解析木を基づいて確実な方法で計算しているため，パラメータの個数が大きくなる傾向にある．そこで，パラメータである j, k の数の削減法を以下に示す．

[削減1] スタックの内容に同じ状態を含んでいる場合，条件となっている状態を削減する．

表 2: 学習した出現確率

$P(u, S_1 \mid S_0) = (N_{uu} + N_{uv})/N$
$P(v, S_2 \mid S_0) = (N_{vu} + N_{vv})/N$
$P(u, S_1 \mid S_0S_1) = N_{uu}/(N_{uu} + N_{uv})$
$P(v, S_2 \mid S_0S_1) = N_{uv}/(N_{uu} + N_{uv})$
$P(x, S_3S_5S_5 \mid S_0S_1S_1) = A_{uu}/N_{uu}$
$P(x, S_3S_5 \mid S_0S_1S_1) = B_{uu}/N_{uu}$
$P(x, S_3S_5S_5S_4 \mid S_0S_1S_1) = C_{uu}/N_{uu}$
$P(x, S_3S_7S_5S_{acc} \mid S_0S_1S_5) = 1$
$P(x, S_3S_7S_5S_5S_{acc} \mid S_0S_1S_1S_5) = B_{uu}/B_{uu} = 1$
$P(x, S_3S_7S_{acc} \mid S_0S_4) = 1$
$P(x, S_3S_6S_5 \mid S_0S_1S_2) = A_{uv}/N_{uv}$
$P(x, S_3S_6 \mid S_0S_1S_2) = B_{uv}/N_{uv}$
$P(x, S_3S_6S_5S_4 \mid S_0S_1S_2) = C_{uv}/N_{uv}$
$P(x, S_3S_7S_6S_5S_{acc} \mid S_0S_1S_2S_6) = B_{uv}/B_{uv} = 1$
$P(u, S_1 \mid S_0S_2) = N_{vu}/(N_{vu} + N_{vv})$
$P(v, S_2 \mid S_0S_2) = N_{vv}/(N_{vu} + N_{vv})$
$P(x, S_3S_5S_6S_3 \mid S_0S_2S_1) = A_{vu}/N_{vu}$
$P(x, S_3S_5 \mid S_0S_2S_1) = B_{vu}/N_{vu}$
$P(x, S_3S_5S_6S_4 \mid S_0S_2S_1) = C_{vu}/N_{vu}$
$P(x, S_3S_7S_6S_{acc} \mid S_0S_2S_6) = 1$
$P(x, S_3S_7S_5S_6S_{acc} \mid S_0S_2S_1S_5) = B_{vu}/B_{vu} = 1$
$P(x, S_3S_6S_6 \mid S_0S_2S_2) = A_{vv}/N_{vv}$
$P(x, S_3S_6S_3 \mid S_0S_2S_2) = B_{vv}/N_{vv}$
$P(x, S_3S_6S_6S_4S_3 \mid S_0S_2S_2) = C_{vv}/N_{vv}$
$P(x, S_3S_7S_5S_6S_{acc} \mid S_0S_2S_2S_6) = B_{vv}/B_{vv} = 1$

例えば，$P(x, S_3S_5S_5 \mid S_0S_1S_1)$, $P(x, S_3S_5 \mid S_0S_1S_1)$, $P(x, S_3S_5S_5S_4 \mid S_0S_1S_1)$ では，$S_0S_1S_1$ が条件になっているが，他の条件付き確率から S_1S_1 に条件を削減しても良い．

[削減 2] 同じ先読み状態を削減する．

$P(x, S_3S_7S_5S_6S_{acc} \mid S_0S_2S_2S_6)$ では，$S_0S_2S_2S_6$ となることがないので，それを削減して $P(x, \phi \mid S_0S_2S_2S_6)$ としても良い．ϕ は空集合を表す．

これらの削減 1,2 を行なうことによって，削減した出現確率を表 3 に示す．表 3 から分かるように，先読み状態個数とスタックの内容を読む個数は $j = 3$, $k = 4$ に削減することができ，よって **4.1** 節の例は PGLR(3, 4) 構文解析であるといえる．

表 3: 学習した出現確率

$P(u, S_1 \mid S_0) = (N_{uu} + N_{uv})/N$
$P(v, S_2 \mid S_0) = (N_{vu} + N_{vv})/N$
$P(u, S_1 \mid S_0S_1) = N_{uu}/(N_{uu} + N_{uv})$
$P(v, S_2 \mid S_0S_1) = N_{uv}/(N_{uu} + N_{uv})$
$P(x, S_3S_5S_5 \mid S_1S_1) = A_{uu}/N_{uu}$
$P(x, S_3S_5 \mid S_1S_1) = B_{uu}/N_{uu}$
$P(x, S_3S_5S_5S_4 \mid S_1S_1) = C_{uu}/N_{uu}$
$P(x, \phi \mid S_1S_5) = (A_{uu} + A_{uv})/(A_{uu} + A_{uv}) = 1$
$P(x, \phi \mid S_1S_1S_5) = B_{uu}/B_{uu} = 1$
$P(x, \phi \mid S_4) = 1$
$P(x, S_3S_6S_5 \mid S_1S_2) = A_{uv}/N_{uv}$
$P(x, S_3S_6 \mid S_1S_2) = B_{uv}/N_{uv}$
$P(x, S_3S_6S_5S_4 \mid S_1S_2) = C_{uv}/N_{uv}$
$P(x, \phi \mid S_1S_2S_6) = B_{uv}/B_{uv} = 1$
$P(u, S_1 \mid S_0S_2) = N_{vu}/(N_{vu} + N_{vv})$
$P(v, S_2 \mid S_0S_2) = (A_{vv} + B_{vv} + C_{vv})/(N_{vu} + N_{vv})$
$P(x, S_3S_5S_6S_3 \mid S_2S_1) = A_{vu}/N_{vu}$
$P(x, S_3S_5 \mid S_2S_1) = B_{vu}/N_{vu}$
$P(x, S_3S_5S_6S_4 \mid S_2S_1) = C_{vu}/N_{vu}$
$P(x, \phi \mid S_0S_2S_6) = (A_{vu} + A_{vv})/(A_{vu} + A_{vv}) = 1$
$P(x, \phi \mid S_2S_1S_5) = B_{vu}/B_{vu} = 1$
$P(x, S_3S_6S_6 \mid S_2S_2) = A_{vv}/N_{vv}$
$P(x, S_3S_6S_3 \mid S_2S_2) = B_{vv}/N_{vv}$
$P(x, S_3S_6S_6S_4S_3 \mid S_2S_2) = C_{vv}/N_{vv}$
$P(x, \phi \mid S_2S_2S_6) = B_{vv}/B_{vv} = 1$

5 おわりに

本稿では，条件付き確率のパラメータを見直すことによって，厳密な確率一般化 LR 構文解析ができることを示した．ただし，パラメータの数が多くなるようなモデルでは，読み込む状態数を減らすことによって近似解を求めることも考えられる．近似解の精度についてはこれからの課題といえる．また，部分解析木の抽出や置換にも利用できるのではないかと考えられる．これについても今後の課題としたい．

参考文献

[1] M. Tomita, *An Efficient Augmented - Context-Free Parsing Algorithm*, Computational Linguistics, 13,1-2, pp31-46, 1987.

[2] T. Briscoe and J. Carroll, *Generalized Probabilistc LR Parsing of Natural Language(Corpora) with Unification-Based Grammars*, Computational Linguistics, Vol.19, No.1, pp.25-59, 1993.

[3] E. Charniak and G. Carroll, *For improved grammatical language models*, proceedings of AAAI-94, pp.728-733, 1994.

[4] E. Charniak, *Statistical langugae learning*, MIT press, 1993.

[5] K. Inui, V. Sornlertlamvanich, H. Tanaka and T. Tokunaga, *Probabilistic GLR Parsing: A New Formalization and Its Impact of Parsing Performance*, Journal of Natural Language Processing, Vol. 5, No. 3, pp.33-52, 1998.

[6] 今井宏樹, 田中穂積, 徳永健伸, 音声認識を目指した確率 GLR 法を用いた言語モデルの構築, 情報処理学会論文誌, Vol 40, No.4, 1404-1412, 1999.