複合化確率文脈自由文法の提案

4AE−6

横林 由理枝　冨浦 洋一　日高 達
(九州大学大学院システム情報科学研究科)

1 はじめに

自然言語処理において, 文脈自由文法 (CFG) ではある文に対して構文解析を行なった場合, 一般に複数の構文木が導出される. そのため解析結果をそのまま意味解析や翻訳等の処理に渡すと処理量が増大する.

そこで, 処理の効率化をはかるために, 構文木間に優先順位を設けて後処理に渡す構文木の数を絞ることが考えられ, 例えば, 構文木に生起確率を与える確率文脈自由文法 (PCFG) を用いて優先順位をつけることができる.

従来の PCFG では標本列 (構文木列) を単一の発生源から収集されたものとしていた. しかし, 実際の言語データが一つの発生源から得られたものか否かわからず, さらに, 自然言語が PCFG で完全に表せるとも限らない. そこで, 実際の言語現象に柔軟に対応できるようにするため, 実際のデータは複数の発生源から収集されるものとして, 複数の PCFG をもつ複合化確率文脈文法 (以下, 複合化文法という) を提案する.

本研究では, 学習データから得られた複合化文法が, 実際にどのような発生源から得られたかがわからないことを考慮して, 文法数 (複合化度) の決定法と適用確率の推定法を提案する.

2 複合化文法とパラメタ推定

【定義 1】 複合化文法 $G^{(M)}$を,PCFG $G_l = \langle G, p_l \rangle$ と選択確率 c_lとの 2 つ組によって定義する.

$$G^{(M)} = \langle \langle \langle G, p_1 \rangle, c_1 \rangle, \cdots, \langle \langle G, p_M \rangle, c_M \rangle \rangle$$

ここで,Mは複合化度とする.

また, それぞれの PCFG G_l は $\langle \langle \Sigma, V, P, X_1 \rangle, p_l \rangle$ で定義される. ただし,

$$\begin{cases} \Sigma & : \text{終端記号の有限集合} \\ V & : \text{非終端記号の有限集合} \\ P & : \text{生成規則の有限集合} \\ & \quad (P = \{\delta_{ij} \mid \delta_{ij} = X_i \to \alpha_j : i = 1, \cdots, n, \\ & \qquad\qquad j = 1, \cdots, m_i\}) \\ X_1 & : \text{開始記号} \ (X_1 \in V) \\ p_l & : P \to (0, 1] \end{cases}$$

である.

また, 選択確率 $c_1, \cdots, c_M$は次の条件が必要である.

$$\sum_{l=1}^{M} c_l = 1 \qquad 0 < c_l \leq 1 \quad (l = 1, \cdots, M)$$

それぞれの文法 G_lにおける p_l $(l = 1, \cdots, M)$ と G_l の選択確率 c_l $(l = 1, \cdots, M)$ の列を次のようなベクトルで表す.

$$\begin{cases} \boldsymbol{p}^{(M)} & = & p_1^{(M)}, p_2^{(M)}, \cdots, p_M^{(M)} \\ \boldsymbol{c}^{(M)} & = & c_1^{(M)}, c_2^{(M)}, \cdots, c_M^{(M)} \end{cases}$$

□

【定義 2】 複合化文法における構文木 T_kの生起確率 $\Pr(T_k \mid \boldsymbol{p}^{(M)})$ は以下のように定義される.

$$\Pr(T_k \mid \boldsymbol{p}^{(M)}) = \sum_{l=1}^{M} c_l^{(M)} \Pr(T_k \mid p_l^{(M)})$$

ただし,$G_l^{(M)}$は複合化文法 $G^{(M)}$の l 番目の PCFG であり, $\Pr(T_k \mid p_l^{(M)})$ $(l = 1, \cdots, M)$ は PCFG $G_l^{(M)}$ における構文木 T_kの生起確率とする.

□

A Proposal of Composite Probabilistic Context Free Grammar
Yurie Yokobayashi, Yoichi Tomiura, Toru Hitaka
Kyushu University
6-10-1 Hakozaki,Higashi-ku,Fukuoka 812-81,Japan

複合化文法における尤度(構文木列の生起確率)は次式から求められる.

$$L_M(\boldsymbol{p}^{(M)}, \boldsymbol{c}^{(M)}) = \prod_{k=1}^{N} \Pr(T_k \mid \boldsymbol{p}^{(M)})$$

この尤度が最大になるようなパラメタ $\boldsymbol{p}^{(M)}, \boldsymbol{c}^{(M)}$ を Baum 理論を用いて推定する. Baum 理論とは, 変数 $\boldsymbol{x}$ の値を更新することによって, 多項式 $f(\boldsymbol{x})$ を極大, あるいは極小にする $\boldsymbol{x}$ の値を求める手法である. 複合化文法における尤度は一般に複数の極大値が存在するため, Baum 理論を用いた推定ではパラメタの初期値によって尤度がどの値に収束するかが左右される.

3 パラメタの初期値設定法

ここでは, パラメタの初期値設定法の考え方を示す.

ここで,Baum 理論において初期値となるものを $p_{l\ init}^{(M)}(\delta_{ij})$, $c_{l\ init}^{(M)}$, 収束計算を行った後の値を $p_{l\ \infty}^{(M)}(\delta_{ij})$, $c_{l\ \infty}^{(M)}$ とする.

ある適当な $p : P \to (0,1]$ に対して

$$\begin{cases} \boldsymbol{p}_{init}^{(M+1)} &= \boldsymbol{p}^{(M)}, p \\ \boldsymbol{c}_{init}^{(M+1)} &= \boldsymbol{c}^{(M)}, 0 \end{cases}$$

とすると,

$$\begin{cases} p_{l\ \infty}^{(M+1)} &= p_l^{(M)} \quad (1 \le l \le M) \\ c_{l\ \infty}^{(M+1)} &= c_l^{(M)} \quad (1 \le l \le M) \end{cases}$$

となり, この場合, 複合化度 M と $M+1$ の文法による尤度は等しくなる. そこで, 上記の $\boldsymbol{p}_{init}^{(M+1)}, \boldsymbol{c}_{init}^{(M+1)}$ の値を少し変えたもの (p の与え方としては, $\boldsymbol{p}^{(M)}$ の要素の中でどれとも傾向がちがうものに設定し, $c_{M+1\ init}^{(M+1)}$ も 0 でない値にする) を初期値として与えてやると, 複合化度 M と $M+1$ の文法による尤度は少なくとも異なることが期待できる. そこで, 複合化度 $M+1$ の文法による尤度が複合化度 M の文法による尤度より大きくなるように, いろいろな変え方で試す.

具体的には複合化の手順を次のように考える.

まず, $c_1 = 1, p_1^1(\delta_{ij})$ を最尤推定法により求めた値に設定する.

次に, 複合化文法 $G^{(M)}$ から $G^{(M+1)}$ を新たにつくる手法を次のように考える.

$$\begin{aligned} p_{l\ init}^{(M+1)}(\delta_{ij}) &= p_l^{(M)}(\delta_{ij}) \quad (1 \le l \le M) \\ p_{M+1\ init}^{(M+1)}(\delta_{ij}) &= \frac{1 - p_{l_0}^{(M)}(\delta_{ij})}{m_i - 1} \\ c_{M+1\ init}^{(M+1)} &= \frac{1}{2} \cdot c_{l_0}^{(M)} \\ c_{l_0\ init}^{(M+1)} &= \frac{1}{2} \cdot c_{l_0}^{(M)} \\ c_{l\ init}^{(M+1)} &= c_l^{(M)} \quad (1 \le l \le M, l \ne l_0) \end{aligned}$$

上式で $l_0 = 1, \cdots, M$ においてすべての l_0 に対して $\boldsymbol{p}_{init}^{(M+1)}, \boldsymbol{c}_{init}^{(M+1)}$ を計算し, それをパラメタの初期値として Baum 理論を用いて収束計算を行ない, 尤度 $L_{M+1}(\boldsymbol{p}_{\infty}^{(M+1)}, \boldsymbol{c}_{\infty}^{(M+1)})$ が最も大きくなるような $\boldsymbol{p}_{\infty}^{(M+1)}, \boldsymbol{c}_{\infty}^{(M+1)}$ をパラメタ $\boldsymbol{p}^{(M+1)}, \boldsymbol{c}^{(M+1)}$ として選ぶ.

以上の繰り返し計算を ϵ が十分小さな正数とすると,

$$L_{M+1}(\boldsymbol{p}^{(M+1)}, \boldsymbol{c}^{(M+1)}) - L_M(\boldsymbol{p}^{(M)}, \boldsymbol{c}^{(M)}) \le \epsilon$$

となるまで行う (このとき, 複合化度は M となる).

4 最後に

今回, 計算機によって発生させた構文木をサンプルにして実験を行なった. 実験の結果はおおむね期待されたものであり, 提案した初期値設定の手法はうまくいったと言える. 今後の課題として, コーパスなどの実際のデータに対して実験を行ない, 構文構造の曖昧性がどのくらい解消されるかを確かめる.

参考文献

[1] 林田 憲昭,「確率文脈自由文法の複合化」, 九州大学大学院修士論文 (1997)

[2] 日高 達,「確率文法」, 情報処理学会誌 Vol.36 No.2(1995)