

proof net を用いたランベック計算の実装に向けて

渡辺成美 (指導教員: 戸次大介)

1 はじめに

自然言語の構文解析として様々な手法が提案されている。Categorial Grammar(CG)では、各語の統語範疇を辞書から引き出し、その統語範疇の列に推論規則を適用することで文を解析する。統語派生を推論とみなし、統語範疇を論理式、辞書を公理の集合、統語規則を推論規則と考えるのである。証明に成功したとき、構文解析ができたこととされる。CGはコンピュータに処理をさせやすいという点で優位である。CGの拡張されたものとしてLambeck Categorial Grammar(LCG)がある。Morrill(2011)はLCGと対応したproof netによる構文解析アルゴリズムを提案している。本研究ではこれに基づいた日本語の構文解析器の実装を目指す。

実装のための言語としてPrologを用いる。論理型言語であるPrologはCGと相性が良いと考えられる。

2 Lambeck Categorial Grammar

ランベック計算の型の集合Fは以下のように定義される:

$$F ::= P \mid F \bullet F \mid F \setminus F \mid F / F$$

LCGではゲンツェン流シーケント計算を行う。/, \、\bulletについてのシーケントを得るための体系が定義されている。規則としてカット規則、恒等規則id、論理規則がある。

$$\begin{array}{c} \frac{}{A \Rightarrow A^{id}} \quad \frac{\Gamma \Rightarrow A \quad \Delta(A) \Rightarrow B}{\Delta(\Gamma) \Rightarrow B} Cut \quad \frac{\Gamma \Rightarrow A \quad \Delta(C) \Rightarrow D}{\Delta(\Gamma, A \setminus C) \Rightarrow D} \setminus L \\ \\ \frac{A, \Gamma \Rightarrow C}{\Gamma \Rightarrow A \setminus C} \setminus R \quad \frac{\Gamma \Rightarrow A \quad \Delta(C) \Rightarrow D}{\Delta(C/B, \Gamma) \Rightarrow D} /L \quad \frac{\Gamma, B \Rightarrow C}{\Gamma \Rightarrow C/B} /R \\ \\ \frac{\Delta(A, B) \Rightarrow D}{\Delta(A, B) \bullet D} \bullet L \quad \frac{\Gamma \Rightarrow A \quad \Gamma \Rightarrow B}{\Gamma, \Delta \Rightarrow A \bullet B} \bullet R \end{array}$$

図1. ゲンツェン流シーケント計算の規則

ランベックシーケント計算においてカット除去が成り立つことを証明されている。また、カット以外の規則はその上段における各型がいずれも下段の部分型であるという性質を持つ。そのため各定理はその部分論理式のみを含む証明を持つ。後ろ向き連鎖証明探索(backward chaining proof search)の際に探索空間が有限であるので、あるシーケントが定理であるかどうかは決定可能である。

単語とその型は辞書の中に定義される。この時生成される言語は有限集合である。

ランベック計算の自然演繹的表現として、以下が与えられる。普通自然演繹に順序の区別はないが、このランベック計算のための自然演繹では順序が存在する。意味論はλ計算により引き出される。シーケント $\Gamma \Rightarrow A$ が導出可能なとき、そしてその時に限り、そのシーケントはランベック計算の定理である。

$$\begin{array}{c} \frac{}{A: \phi} \quad \frac{}{A \setminus B: \chi} E \setminus \quad \frac{}{A: \phi} \quad \frac{}{A \setminus B: \lambda x \psi} I \setminus \quad \frac{}{B/A: \chi} \quad \frac{}{A: \phi} E / \quad \frac{}{B: (\chi \phi)} E / \end{array}$$

$$\frac{}{A: x^i} \quad \frac{}{A: \phi} \quad \frac{}{A \setminus B: (\phi, \psi)} I \bullet$$

図2. 意味論を含めたランベック計算のための自然演繹

3 Proof net

proof netでは証明を図で表記する。木 $|A^P|$ は、図3の変換に従って定義される、型 A^P の木である。

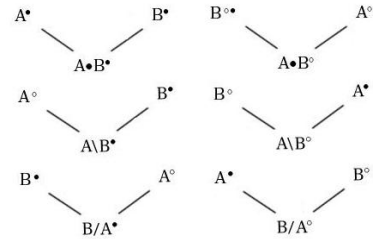


図3. proof netの木への変換関数

proof frameはシーケント計算と対応する。ただしランベック計算が問題として抱えている擬似的曖昧性が、proof netには存在しない。シーケント $A_1, \dots, A_n \Rightarrow A_0$ は $(|A_0^\bullet|, |A_1^\bullet|, \dots, |A_n^\bullet|)$ で表される。

A^\bullet と A° は相補的であり $(\overline{A^\bullet} = A^\circ, \overline{A^\circ} = A^\bullet)$ 、その間にリンクを結ぶことができる。このリンクは互いに交差ししない。また、/° や \° のから伸びる葉のうち、最左の葉と最右の葉を繋げることはできない。リンクが全て繋げられ、proof netが正しく形成されるとき、そのシーケントはランベックシーケント計算で導出可能な定理である。

例として、「太郎が手紙を書く」の構文解析を行う。戸次(2010)に基づいて辞書を以下のように定義する:

太郎	: NP_{nc}	:	taro
手紙	: NP_{nc}	:	tegami
が	: $NP_{nc} \setminus NP_{ga}$:	id
を	: $NP_{nc} \setminus NP_o$:	id
書	: $NP_o \setminus (NP_{ga} \setminus S_{v,5,k,stem})$:	$\lambda x \lambda y (kaku(x, y))$
く	: $S_{v,5,k,stem} \setminus S_{v,5,k,term}$:	id

この文に一致するシーケントは、
 $NP_{nc}, NP_{nc} \setminus NP_{ga}, NP_{nc}, NP_{nc} \setminus NP_o,$
 $NP_o \setminus (NP_{ga} \setminus S_{v,5,k,stem}), S_{v,5,k,stem} \setminus S_{v,5,k,term} \Rightarrow S_{term}$
 である。このシーケントの導出は図4に与えられる。

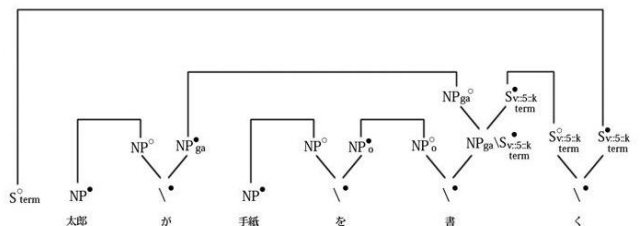


図4. 「太郎が手紙を書く」の導出

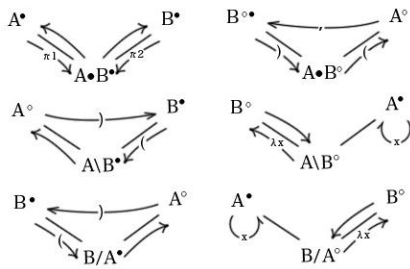


図 5.semantic trip

意味論は図 5 に従って A_0° から順に移動していくことで引き出される。 A_0° に戻ること終了する。図 4 の場合 (1a) が導かれ、(1b) に縮約される。

- (1) a. $(id((\lambda x \lambda y(kaku(x, y))(id\ tegami))(id\ taro)))$
 b. $(kaku(tegami\ taro))$

4 proof net を用いた prolog による実装

上記の構文解析アルゴリズムを用いたプログラムが Morrill(2011) により提供されている。そのアルゴリズムは以下の通りである：

$$\frac{\frac{\text{process } \alpha}{[S_0^\circ], [], \alpha, [S_0^\circ], \emptyset}^{START} \quad \frac{G, [L_i | L_s], \alpha, F, X}{[L_i | G], L_s, \alpha, F, X}^{SHIFT}}{G, [], \alpha \oplus \alpha', F, X} \quad \frac{G, \text{fringe}([A^\circ |_{\#(F)}], \alpha', F \oplus [A^\circ |_{\#(F)}], X)}{LEX, \text{where } \alpha : A}$$

$$\frac{[], [], F, X}{\text{accept}}^{END} \quad \frac{[L_i | G], [L_j | L_s], \alpha, F, X}{G, L_s, \alpha, F, X \cup \{(i, j)\}}^{REDUCE}$$

ただし L_i と L_j は相補的であり、frame F において subtend せず、 F にあるもの間の各 path は axiom links X と共にいくつかの ϕ -link の edges のうち両方を横切る。

- G : メインスタック。まだ試されていない葉を含む
- L_s : 補助スタック。
- α : まだ見られていない単語
- F : 語彙選択によってこれまでに立てられた proof net
- X : proof net 上にこれまでに立てられたリンクの集合

図 6. 構文解析アルゴリズム解析

このプログラムでは型に変数が存在せず、そのため例えば ‘and’ の辞書を以下のように記述する必要がある：

- and : $(NC \setminus NC) / NC : \lambda x \lambda y [(y \wedge x)]$
 and : $((N \setminus S) \setminus (N \setminus S)) / (N \setminus S)$
 : $\lambda x \lambda y \lambda z [(y z) \wedge (x z)]$
 and : $((((N \setminus S) / N) \setminus ((N \setminus S) / N)) / ((N \setminus S) / N))$
 : $\lambda x \lambda y \lambda z \lambda w [(y z w) \wedge (x z w)]$
 ...

$(T \setminus T) / T$ の形を取るものが数多く並び、冗長である。また日本語の構文解析を考えた時、助詞を扱うのと同じ問題が挙がる可能性がある。

これを踏まえて、本研究では型に変数を用いることができるようにした日本語構文解析器を実装した。図 6 の LEX の適用の際、単語 A に含まれる変数 T_1, \dots, T_n に、他の単語の変数と重ならないように番号を付けなおす。REDUCE で変数が他の型と結びつけられたとき、 L_s および G 中のそれと同じである変数は書き換えられる。このようにして変数が処理される。

素性は型の中に含めて扱う。このとき格助詞「が」

及び並列の助詞「と」は以下のように表される：

- が : $(NP \setminus T_1) / (NP_{ga} \setminus T_1) : id$
 と : $(T_1 \setminus T_1) / T_1 : \lambda x \lambda y [(x \wedge y)]$

「が」の T_1 と「と」の T_1 はプログラム中では異なる変数として識別される。図 7 では T_1 に NP_{nc} 、 T_2 に S_{term} が代入されている。「太郎と次郎が走る」の意味論として (2a) が導かれ、これは (2b) に縮約される。

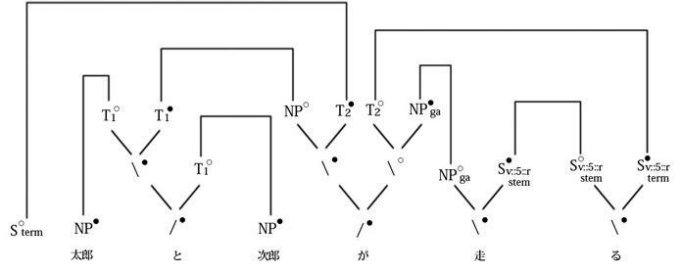


図 7. 「太郎と次郎が走る」の導出

- (2) a. $((id(\lambda y(id(\lambda x(hashiru\ x)\ y))))$
 $((\lambda z \lambda w(z \wedge w)\ jiro)\ taro))$
 b. $(hashiru(taro \wedge jiro))$

5 まとめと今後の課題

proof net による構文解析アルゴリズムを用いて、日本語の解析を行うことができた。現在日本語のかき混ぜ規則に対応していない。「太郎が手紙を書く」と「手紙を太郎が書く」の両方を解析するには、辞書に以下の二つがなければならない：

- 書 : $NP_o \setminus (NP_{ga} \setminus S_{v,5,k,stem}) : \lambda x \lambda y (kaku(x, y))$
 書 : $NP_{ga} \setminus (NP_o \setminus S_{v,5,k,stem}) : \lambda x \lambda y (kaku(x, y))$

上記の proof net ではかき混ぜ規則を表現することができないため、拡張する必要がある。proof net のリンクは互いに交わらないと書いたが、条件付きで交わることを許すことにより、かき混ぜ規則を実現できるかもしれない。また現在素性に対する扱いが弱く、変数を入れることができない。日本語の活用柔軟に対応するには、素性についてより正確に記述できる構造にする必要がある。

6 参考文献

- [1] Glyn V. Morrill (2011) *Categorical Grammar Logical Syntax, Semantics, and Processing*, Oxford
- [2] 戸次大介 (2010) 日本語文法の形式理論 - 活用体系・統語構造・意味合成、くろしお出版