CPS 変換と多相範疇文法

谷口雅弥[1]

December 3, 2020

概要

研究背景・研究手法・先行研究

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced past

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced past the

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced past the barn

研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced past the barn fell.

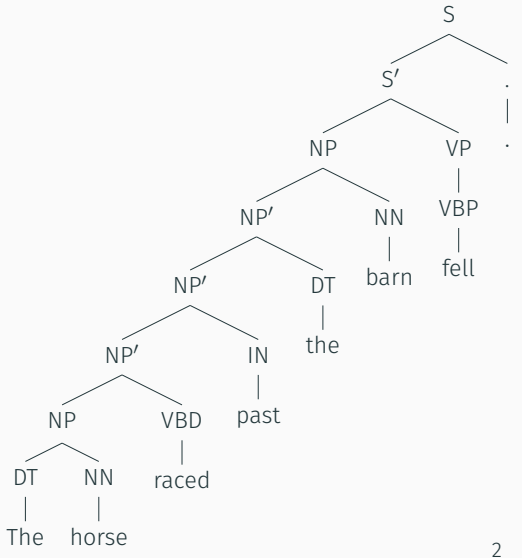
研究背景: 漸進的構文解析

漸進的構文解析

自然言語の文を分析するとき文頭から単語を追って文を分析する場合に，再分析が発生する例を**袋小路文** (garden-path sentence) という．文章を文頭から解析する技術は同時通訳などへの応用が考えられる．

例文

The horse raced past the barn fell.



研究手法: 組み合わせ範疇文法

定義 (組み合わせ範疇文法 (combinatory categorial grammar))
次の 2 つの型を単語ごとに割り当てた語彙をもとに 与えられた文に対して β 簡約や関数合成などを行う．与えられた計算結果の構文型が S となるときに，その文は受理される．

構文型 統語範疇 (*VP*, *NP* など) に関数型 (X/Y と $X\backslash Y$) を加えたもの

意味型 単純型付きラムダ計算の項

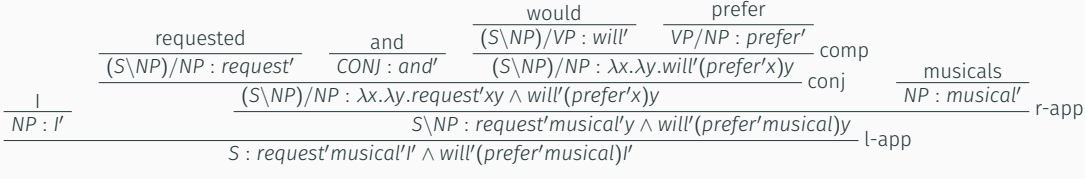
研究手法: 組み合わせ範疇文法

定義 (組み合わせ範疇文法 (combinatory categorial grammar))
次の 2 つの型を単語ごとに割り当てた語彙をもとに 与えられた文に対して β 簡約や関数合成などを行う．与えられた計算結果の構文型が S となるときに，その文は受理される．

構文型 統語範疇 (VP, NP など) に関数型 (X/Y と $X\backslash Y$) を加えたもの

意味型 単純型付きラムダ計算の項

例
“I requested and would prefer musicals.” は次のように構文解析される．



先行研究: 組合わせ範疇文法に基づく漸進的な意味解析

先行研究
Kato and Matsubara, 2014 は組合せ範疇文法に接合操作を導入することで漸進的に構文木を構成した．

問題点
部分構文木に対して，新たに接合する木が接合可能か否かを判定する方法は組合わせ範疇文法で定義せず，木接合文法に基づいており意味解析の表現方法として組合せ範疇文法を用いている．つまり，木接合文法に基づいて構文木を構成し，その構文木を組合せ範疇文法として解釈し直すことで意味解析を行っている．

本研究の目的
構文解析自体も組合せ範疇文法だけで行えるように，文法規則を拡張する．こうすることで意味解析と構文解析の一体化を行う．

予備知識

予備知識: 多相ラムダ計算 λ_2

Girard の多相ラムダ計算 λ_2 は単純型付きラムダ計算の拡張であり，次のように定義される．

定義 (型 τ_2)
α を原始的な型として，β を型変数とする．このとき型 τ_2 は次のよう定義される．

$$\tau_2 ::= \tau_2 \to \tau_2 \mid \alpha \mid \beta \mid \Pi\beta.\tau_2$$
$$\alpha ::= a \mid b \mid \cdots$$
$$\beta ::= \mathrm{U} \mid \mathrm{V} \mid \cdots$$

括弧'()' は $\to$ の優先順位を示すために予約されている．

予備知識: 多相ラムダ計算 λ_2

定義 (定数と変数の型)
型のついていない変数の無限列を仮定する．このとき型付き変数を次のように定義する．

- 一つ以上の型と対応する型のついていない変数は存在しない．
- どの型もある型ついていない変数の無限列と対応する．

x を変数として τ を τ_2 の型とする．そのとき型付き変数は x^τ と表記する．また一つの型対して原始的な定数の有限もしくは無限の列が存在すると仮定する．このとき，c を原始的な定数として c^τ と表記する．

予備知識: 多相ラムダ計算 λ_2

定義 (型付きラムダ項)
型付きラムダ項は次の条件から定義される.

- すべての型付きの原始的な定数と変数はラムダ項である.
- τ と σ を τ_2 の型する. x^τ を型付変数として, M^σ が型付きラムダ項とする. このとき $(\lambda x^\tau.M^\sigma)^{\tau\to\sigma}$ は型付きラムダ項である.
- τ と σ を τ_2 の型とする. $M^{\tau\to\sigma}$ と N^τ を型付きラムダ項とする. このとき, $(M^{\tau\to\sigma}N^\tau)^\sigma$ は型付きラムダ項である.
- X を型変数とする. そして M^τ は型付きラムダ項とする. このとき $(\Lambda\mathrm{X}.M^\tau)^{\Pi\mathrm{X}.\tau}$ は型付きラムダ項である.
- σ は型変数とする. そして $(\Lambda\mathrm{X}.M^\tau)^{\Pi\mathrm{X}.\tau}$ を型付きラムダ項とする. このとき $\left((\Lambda\mathrm{X}.M^\tau)^{\Pi\mathrm{X}.\tau}\sigma\right)^{[\sigma/\mathrm{X}]\tau}$ は型付きラムダ項である.

予備知識: Continuation-Passing Style 変換

定義 (Continuation-Passing Style 変換)
以下に単純型付きラムダ計算における Continuation-Passing Style 変換を示す．このとき，変換後の Answer Type として多相ラムダ計算の型変数 X を用いる. 型を右肩に乗せると煩雑になるため省略し，その型の変換は別記する．

$$\begin{aligned}
\overline{x} &= \Lambda X.\lambda\kappa.\kappa x \\
\overline{\lambda x.M} &= \Lambda X.\lambda\kappa.\kappa(\lambda x.\overline{M}) \\
\overline{MN} &= \Lambda X.\lambda\kappa.\overline{M}(\lambda\mu.\overline{N}(\lambda\nu.\mu\nu\kappa))
\end{aligned}$$

予備知識: Continuation-Passing Style 変換

定義 (Continuation-Passing Style 変換)
このとき，ラムダ項の型 ω は次のように $\overline{\omega}$ として定義される．α は原始的な型として τ, σ は τ_2 の任意の型とする．

$$
\begin{aligned}
\overline{\tau} &= \Pi X.T(\tau^*) \\
T\tau &= (\tau \to X) \to X \\
\alpha^* &= \alpha \\
(\tau \to \sigma)^* &= (\tau^* \to T(\sigma^*))
\end{aligned}
$$

研究内容

Type-raising 規則

関数の適用と合成だけでは自然言語を十分に構文解析出来ないため，あらたに Type-raising 規則を導入する．

定義 (Type-raising 規則)
任意の名詞句 $NP : a$ は，次のように繰り上げられる．

$$\frac{NP : a}{\Pi T.T/(T\backslash NP) : \Lambda T.\lambda f^{T\backslash NP}.fa}$$

$$\frac{NP : a}{\Pi T.T\backslash(T/NP) : \Lambda T.\lambda f^{T/NP}.fa}$$

推論規則としての Continuation Passing Style 変換

Taniguchi and Tojo, 2020 は型の繰り上げ規則の一般化として Continuation Passing Style 変換を組合せ範疇文法に導入した.

定義 (Continuation-Passing Style 変換)

$$\frac{U/V\$: f}{\Pi X.(X/V\$)/(X\backslash U) : \Lambda X.\lambda k^{X\backslash U}.\lambda v^{V}\$.k(fv\$)}\ /\mathrm{CPS}$$

$$\frac{U\backslash V\$: f}{\Pi X.(X\backslash V\$)\backslash(X/U) : \Lambda X.\lambda k^{X/U}.\lambda v^{V}\$.k(fv\$)}\ \backslash\mathrm{CPS}$$

組み合わせ範疇文法に基づく漸進的構文解析

補題 (関数合成を右適用に変換)
任意の関数合成は右適用の形に変換することができる.

$$\frac{X/Y : f \qquad Y/Z : g}{X/Z : \lambda a.g(fa)}\ \text{comp}$$

$$\frac{X\backslash Y : f \qquad Y\backslash Z : g}{X\backslash Z : \lambda a.g(fa)}\ \text{comp}$$

$$\frac{\dfrac{X/Y : f}{(X/Z)/(Y/Z) : \lambda h.\lambda a.h(fa)}\ \text{cps} \qquad Y/Z : g}{X/Z : \lambda a.g(fa)}\ \text{r-app}$$

$$\frac{\dfrac{X\backslash Y : f}{(X\backslash Z)/(Y/Z) : \lambda h.\lambda a.h(fa)}\ \text{cps} \qquad Y\backslash Z : g}{X\backslash Z : \lambda a.g(fa)}\ \text{r-app}$$

補題 (左適用を右適用に変換)

$$\frac{Z : a \qquad Y\backslash Z : f}{Y : fa}\ \text{l-app}$$

$$\frac{\dfrac{Z : a}{Y/(Y\backslash Z) : \lambda f.fa}\ \text{cps} \qquad Y\backslash Z : f}{Y : fa}\ \text{r-app}$$

組み合わせ範疇文法に基づく漸進的構文解析

これらの補題を利用することで，任意の組合せ範疇文法の導出木を右適用と Continuation Passing Style だけの形に変換することができる．

$$\dfrac{\dfrac{\dfrac{\dfrac{\text{I}}{NP : I}}{(S/NP)/((NP\backslash S)/NP) : I'}\ \text{cps} \quad \dfrac{\text{love}}{(NP\backslash S)/NP : love'}}{S/NP : \lambda x.love'xI'}\ \text{r-app} \quad \dfrac{\text{you}}{NP : you'}}{S : love'you'I'}\ \text{r-app}$$

研究結果・考察・課題

考察: 文脈情報としての Continuation Passing Style

- 導出木の部分木は人間が文頭から文を受け取って行ったときに知っている知識とみなすことができる
- Continuation Passing Style 変換して受け取った継続というのは，今後来るであろうと人間が期待している残りの文の統語範疇とみなすことができる．
- 　つまり統語範疇にも文脈情報が乗っている

課題: 構文の生成能力

- Continuation Passing Style 変換を推論規則として導入すると，文法の能力がどれだけ拡張されてしまうのがわからない
- 推論規則が増えれしまったので組み合わせ範疇文法として試すべき規則が増えたため，バックトラッキングが発生することが増え，計算量が増加している．