

CCG を用いた Spatial Description Clause の拡張への取り組み

稲子 明里† 塚原 裕史‡ 小林 一郎†
 †お茶の水女子大学 ‡株式会社 デンソー アイティラボラトリ

1 はじめに

近年, 車の自動運転の実用化に向けた動きが活発化している. 今後, 車を運転できない人でも自動運転車を操作するために, 口頭指示による容易な操作が必要になることが予想される. そこで本研究では, 自然言語で表現された操作指示内容と実世界との対応づけ (グラウンディング) を目的とし, 駐車指示内容を車の操作へと結びつける空間意味記述へ変換する手法を提案する.

2 空間意味記述を用いたグラウンディング

本研究では, 指示内容から適切な駐車位置を推定するために, 空間の特徴を Kollar ら [1] によって提案された Spatial Description Clause(SDC) を用いて表現する. また, 指示内容と観測した物体とのグラウンディングには, SDC の構造に従った確率的グラフィカルモデルである Generalized Grounding Graphs(以下, グラウンディンググラフ) を生成するためのアルゴリズム [2] を拡張して用いる.

2.1 Spatial Description Clause

Spatial Description Clause(SDC)[1] とは, figure, relation, landmark の 3 要素から構成される空間意味記述のことである. 空間表現を含む言語指示は SDC の木として表現できる. また各 SDC は Event, Object, Place, Path の 4 つのタイプに分けられる. Event は実世界で実行される行動のシーケンス, Object は自動車や人間などの物体, Place は場所, Path は経路や方向を表す.

2.2 グラウンディンググラフ

グラウンディンググラフでは, 言葉とグラウンディングの対象可能な組み合わせから全体の尤度が最大になるものが探索され, 以下の式で表される.

$$\Gamma^* = \arg \max_{\Gamma} p(\Phi = True | SDCs, \Gamma) \quad (1)$$

Γ はグラウンディング γ_i の集合, Φ は言葉がグラウンディングと一致していたら True を返す二値関数 ϕ_i の集合, $SDCs$ は言語指示から変換された SDC である.

An Extension of Spatial Description Clause Using Combinatory Categorical Grammar
 †Akari INAGO ‡Hiroshi TSUKAHARA †Ichiro KOBAYASHI
 †Ochanomizu University ‡Denso IT Laboratory, Inc.

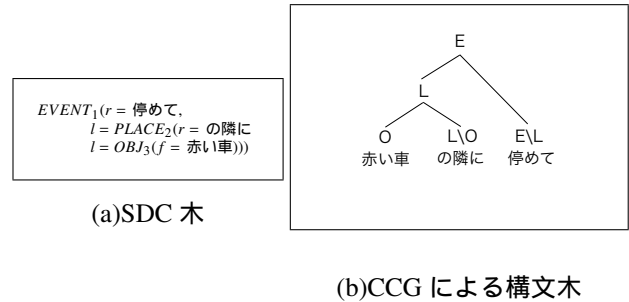


図 1: SDC 木と CCG による構文木

3 CCG を用いた SDC 木への変換

3.1 Combinatory Categorical Grammar

Combinatory Categorical Grammar(CCG)[3] とは, カテゴリに対する関数適用により構文解析を行う範疇文法である. CCG では, カテゴリとして N(名詞) や NP(名詞句), S(文) が存在し, 演算子 “\”, “/”, “/” を組み合わせる. また CCG は, 関数適用規則と組み合わせ規則という 2 つの規則をもつ. 本研究では, SDC のタイプに従い E(Event), O(Object), L(Place), P(Path) を基本的なカテゴリとし, $X/Y \Rightarrow X$ と $YX \Rightarrow X$ の 2 つの関数適用規則のみを用いる. 図 1 に「赤い車の隣に停めて」という駐車指示を SDC 木に変換した例と CCG による構文木に変換した例を示す.

3.2 Shift-Reduce 法に基づくパーサの構築

Shift-Reduce 法とは, スタックを利用して深さ優先探索でボトムアップの構文解析を行う手法である. Shift-Reduce 法では, スタックに対して 2 つの操作 (アクション) を行う. 1 つは, 入力文から単語を一つ取り出してスタックにプッシュする Shift と呼ばれる操作, もう 1 つは, スタックの先頭に文法規則の右辺と一致する単語があるときにそれらを文法規則の左辺のカテゴリに変換する Reduce とよばれる操作である. 本研究では, 文法規則に CCG を用いた Shift-Reduce 法に基づくパーサを構築する.

3.3 未知語の処理

文法規則の右辺にない単語がある場合, 条件付確率場を用いてカテゴリの推定を行う. 使用する素性は, 前後 2 文字の単語, 品詞細分類, カテゴリである. 予測されたカテゴリに従って構文解析できる場合は, そのカテゴリ

を単語とともに文法規則に追加することで未知語の処理を行う。

3.4 パーサのデコーディング

Algorithm1 にパーサの手順を示す。まず、未知語の処理を行い、元の文法規則と照らし合わせて未知語がある場合は、新たな規則を文法規則に追加する。その後、入力文の形態素解析を行いキューに入れ、Shift-Reduce法により構文解析を行う。デコーディングにはビームサーチを用いる。Algorithm1 において、item はスタックとキューの状態構成され、初期状態には空のスタックと入力文の形態素が入ったキューが入る。各ステップで文法規則に従ってアクションを適用し、Reduce 時のスコアのみを対象に累加する。スタックにカテゴリのみが残り、キューが空になった場合に構文解析終了とし、累加したスコアを Reduce の回数で割りスコアとして採用する。

Algorithm 1 パーサの処理手順

```

function: DECODE(input, agenda, list, N, grammer,
                 candidate_output):
    agenda.clear()
    agenda.insert(GETSTARTITEM(input))
    candidate_output = NONE
    while !agenda.empty() do
        list.clear()
        for action to grammer.getActions(item) do
            new_item = item.apply(action)
            if new_item.F == true then
                new_item.score = new_item.score / reduce_num
                if candidate_output == NONE or
                new_item.score > candidate_output.score
                then
                    candidate_output = new_item
                else
                    list.append(new_item)
                end if
            end if
        end for
        agenda.clear()
        agenda.insert(list.best(N))
    end while
    
```

3.5 スコアモデル

候補の item のスコアとして、アクションの識別器により算出した事後確率の対数尤度の和をとる。素性は各アクションを適用した際の item の状態からとり、識別器にはロジスティック回帰を用いる。表 1 はパーサに使用される素性の型を示している。S₀, S₁, S₂, S₃ はスタックの先頭 4 個のノードを表し、Q₀, Q₁, Q₂ はキューの先頭 3 個のノードを表す。p, w, c はそれぞれ品詞, 単語, カテゴリを表す。

4 実験

4.1 実験設定

対象データには、クラウドソーシングによって集められた 10 のシーンにおける 4,865 文の駐車指示を用いた。SDC で表現した際に Path のタイプを含む文は除いた。10 分割-交差検

表 1: 素性の型

型	素性内容
1	S ₀ pw, S ₀ c, S ₀ cw, S ₁ pw, S ₁ c, S ₁ cw, S ₂ cw, S ₃ cw
2	Q ₀ pw, Q ₁ pw, Q ₂ pw
3	S ₀ cwQ ₀ pw, S ₀ cQ ₀ pw, S ₀ cwQ ₀ p, S ₀ cQ ₀ p, S ₁ cwQ ₀ pw, S ₁ cQ ₀ pw, S ₁ cwQ ₀ p, S ₁ cQ ₀ p
4	S ₀ cwS ₁ cQ ₀ p, S ₀ cS ₁ cwQ ₀ p, S ₀ cS ₁ cQ ₀ pw, S ₀ cS ₁ cQ ₀ p, S ₀ pS ₁ pQ ₀ p, S ₀ cwQ ₀ pQ ₁ p, S ₀ pQ ₀ pQ ₁ p, S ₀ cQ ₀ pwQ ₁ p, S ₀ pQ ₀ pQ ₁ pw, S ₀ cQ ₀ pQ ₁ p, S ₀ cwS ₁ cS ₂ c, S ₀ cS ₁ cwS ₂ c, S ₀ cS ₁ cS ₂ cw, S ₀ cS ₁ cS ₂ c, S ₀ pS ₁ pS ₂ p

表 2: パーサの精度

未知語処理	全体の正解率	構文解析できた文の割合	構文解析できた文の正解率
あり	0.551	0.628	0.877
なし	0.560	0.621	0.902

定を行い、未知語の処理を行ったパーサと行っていないパーサの精度を比較する。

4.2 実験結果と考察

精度の結果を表 2 に示す。未知語の処理を行わない場合に比べ行った方が精度は下がるが、構文解析できた割合が上がった。構文解析できた割合が上がったのは、未知語の文法規則を追加したことにより、元の文法規則だけでは構文解析不可能だった文が解析できるようになったことによる。また精度が下がったのは、未知語の処理の予測精度の低さや構文解析後に複数の構文木の解が求まった場合に間違った解のスコアが高くなってしまったことが原因だと考えられる。構文木生成に成功した事例がまだ少ないことについては、CCG の規則として 2 つの関数規則しか用いていないため、規則適用の制約が強すぎる事が考えられる。

5 おわりに

本研究では操作指示文に対する SDC を得る前処理として、文における空間的意味の依存関係が構文木として表示可能な CCG による文法を定義し、この構文木から SDC への変換を行う手法を提案した。さらに、Shift-Reduce 法を用いて定義した CCG に基づくパーサの実装を行った。今後は、パーサの精度の向上、操作指示が持つ条件や視点を含む、より複雑な文を解析できる文法規則の追加や CCGBank の拡充、及びグラウンディンググラフの構築を目指す。

参考文献

- [1] Thomas Kollar, Stefanie Tellex, Deb Roy and Nicholas Roy “Toward Understanding Natural Language Directions”, In Proc., ACM/IEEE Int ’11 Conf. on Human-Robot Interaction (HRI), 259-266, 2010.
- [2] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R. Walter, Ashis Gopal Banerjee, Seth Teller and Nicholas Roy “Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation”, In Proc. AAAI, 2011.
- [3] Mark Steedman. 2000. The Syntactic Process. The MIT Press, Cambridge, Mass.
- [4] Yue Zhang, Stephen Clark “Shift-Reduce CCG Parsing”, In Proc. ACL 2011, pages 683-692, Portland, OR, 2011.