

空間意味記述を対象にした CCGに基づく Shift-Reduce パーザの開発

稲子明里[†]

[†] お茶の水女子大学

塚原裕史[‡]

[‡] 株式会社デンソーアイティラボラトリ

小林一郎[†]

[†]{g1320504, koba}@is.ocha.ac.jp, [‡]htsukahara@d-itlab.co.jp

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

2.1 関連研究

空間意味記述を用いたグラウンディングの手法として, Kollar ら [1] は Spatial Description Clause(SDC) という意味構造を提案した. Tellex ら [2] は SDC の構造に従った確率的グラフィカルモデル(グラウンディンググラフ)を生成するアルゴリズムである Generalized Grounding Graphs を開発し, 空間情報を含む自然言語命令を理解するためのシステムを構築した. Kollar らが SDC を系列ラベリング問題として近似し条件付き確率場 (CRF) で扱う一方, 我々は, 自然言語文の構文情報により, 与えられた言語指示に対する SDC のラベルを正解に付与することができると考え, 構文解析に対して SDC のタイプにより定義された CCG を文法とする Shift-Reduce パーサを構築する. また, そこで得られた結果を SDC への変換規則を用いることによりグラウンディンググラフを生成する.

2.2 提案手法

本研究では言語指示に対する SDC を生成する前処理として, 文における空間的意味の依存関係が構文木として表示できるような 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. 指示内容と観測した物体とのグラウンディングには, Generalized Grounding Graphs[2] を拡張して用いる. 図 1 に提案手法の概要を示す.

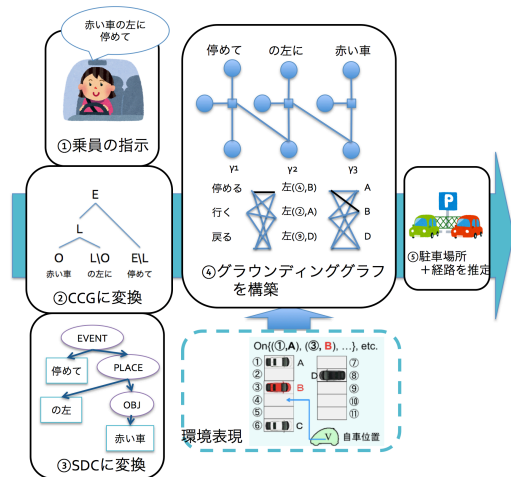


図 1: 概要

3 CCG を用いた SDC 木への変換

3.1 Spatial Description Clause

SDC 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. 葉の SDC は figure のみを含み, 内部節の SDC は他の要素も含む. 内部節の SDC の figure と landmark はそれ自身が SDC となる. 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.

3.2 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。

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

Γ はグラウンディング γ_i の集合、 Φ は言語指示の一部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、 $commands$ は自然言語による指示文である。グラウンディングは物体の位置、バウンディングボックス、ラベルのリストで表される。式 (1) の確率分布はファクタ Ψ_i と正規化項 Z を用いて以下のように変形される。

$$\begin{aligned} p(\Phi | commands, \Gamma) &= p(\Phi | SDCs, \Gamma) \\ &= \frac{1}{Z} \prod_i \Psi_i(\phi_i, SDC_i, \Gamma) \quad (2) \end{aligned}$$

$SDCs$ は言語指示から変換された SDC である。グラフは確率変数 (ϕ, γ, λ) とファクタという 2 つのタイプのノードから構成される。各確率変数は 1 つかそれ以上のファクタノード Ψ_i に接続する。ファクタ Ψ は 2 つのタイプに分けられる。1 つは、葉の SDC の構造からとられる $\Psi(\phi_i, \lambda_i^f, \gamma_i)$ 、もう 1 つは内部節の SDC による $\Psi(\phi_i, \lambda_i^r, \gamma_i^f, \gamma_i^{l1}, \gamma_i^{l2})$ である。図 3 は「赤い車の隣に停めて」という駐車指示のグラウンディンググラフの例である。

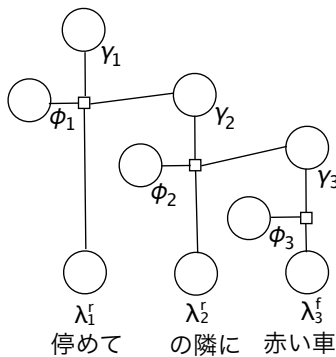


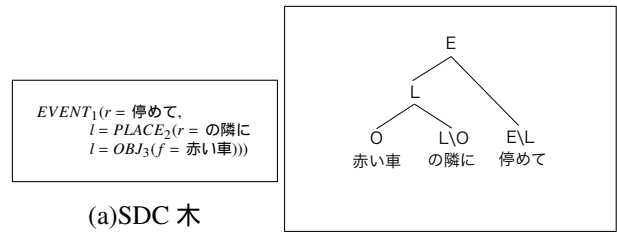
図 2: グラウンディンググラフの例

4 CCG を用いた SDC 木への変換

4.1 Combinatory Categorical Grammer

Combinatory Categorical Grammer (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語彙化文法の 1 つである。CCG のカテゴリは、基底カテゴリと、それらを演算子 “\”, “/” で組み合わせて再帰的に定義される関数型カテゴリから成る。また CCG は、関数適用規則のような組み合わせ規則をもつ。本研究では、SDC のタイプに従い E(Event), O(Object), L(Place),

P(Path) を基底カテゴリとし、 $X/Y \Rightarrow X$ と $Y \backslash X \Rightarrow X$ の 2 つの関数適用規則のみを用いる。図 3 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。



(a) SDC 木

(b) CCG による構文木

図 3: SDC 木と CCG による構文木

4.2 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左辺と一致する単語があるときにそれらを文法規則の右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。図 4 は Shift-Reduce 法による構文解析の例である。

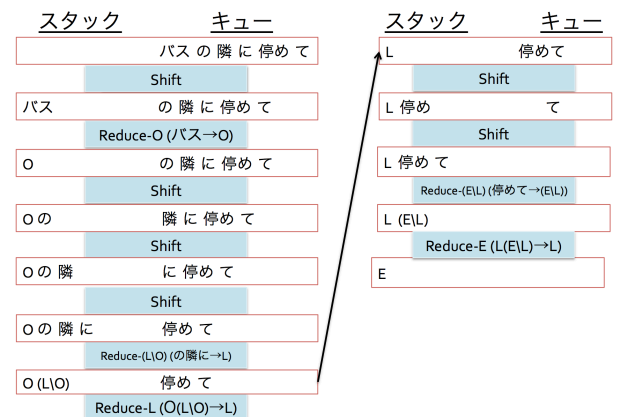


図 4: Shift-Reduce 法による構文解析

4.3 未知語の処理

辞書にない単語がある場合、CRF 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。

4.4 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で構成され、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 (4.5 に詳述) 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。最も高いスコアを持つ item を解とする。

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

```

4.5 スコアモデル

候補の item のスコアとして適用されたアクションに対する確率の対数の Shift を除くアクションの和をとる。素性は各アクションを適用した際の item の状態からとり、確率はロジスティック回帰で学習する。表 1

表 1: 素性の型

型	素性内容
1	$S_0pw, S_0w, S_1pw, S_1c, S_1cw, S_2cw, S_3cw$
2	Q_0pw, Q_1pw, Q_2pw
3	$S_0wS_1w, S_0wS_1c, S_0wQ_0pw, S_0wQ_0p, S_1cwQ_0pw, S_1cQ_0pw, S_1cwQ_0p, S_1cQ_0p$
4	$S_0wS_1cQ_0p, S_0wS_1cwQ_0p, S_0wS_1cQ_0pw, S_0pS_1pQ_0p, S_0wQ_0pQ_1p, S_0pQ_0pQ_1p, S_0wQ_0pwQ_1p, S_0pQ_0pQ_1pw, S_0wS_1cS_2c, S_0wS_1cwS_2c, S_0wS_1cS_2cw, S_0pS_1pS_2p$

は識別器構築に使用される素性の型を示している。 S_0, S_1, S_2, S_3 はスタックの先頭 4 個のノードを表し、 Q_0, Q_1, Q_2 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。

5 実験

5.1 スコアモデルの精度の測定

本研究では、まずパーサのアクションのスコアモデルの精度を以下に示す。

5.1.1 実験設定

ロジスティック回帰の学習には、LIB-linear* を利用した。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コンビニエンスストアの駐車場で駐車シーンにおける 4,865 文の駐車指示を用いた。これらの文はランドマークを含まない指示、ランドマークを 1 つ含む指示、ランドマークを 2 つ含む指示に分けられる。駐車可能なスペースの数は 3 つである。SDC で表現した際に Path のタイプを含む文は事例数が少ないため、今回は除いた。表 2 に収集した文を構文解析して割り当てたカテゴリの組み合わせによる分類を示した。駐車指示は全部で 39 種類存在する。交差検定を行い、アクション毎に精度を比較する。

5.1.2 実験結果と考察

精度の結果を表 3 に示す。ここで、アクションの S は Shift, R-X はカテゴリ X での Reduce である。出現頻度が低いアクションに対する F 値は、低い傾向にあるということがわかった。F 値が低いアクションが行われる文はあまり使われない言い回しを含む文であると仮定し、文を確認したところ、“目の前の真ん中の間”などのランドマークを示す単語が省略された表現が多いことがわかった。また、F 値が極端に低いアクションの

*<http://www.csie.ntu.edu.tw/~cjlin/liblinear/>

表 2: 収集した車を駐車する際に発する 39 種類の自然言語指示の一部

	CCG による構文解析パターン	例文	解析精度
1	$(O/O) + O + (E/O)$	右側の空いている所に停めて	0.929
2	$O + (E/O)$	コンビニ入り口に停めて	0.923
3	$O + (L/O) + (E/L)$	自車の右側に停めて	0.796
...
38	$L + (O/O)\backslash L + O$	右から 2 台目の場所	0
39	$O + (O/O)\backslash O + O + (O/O)\backslash O + O + (L/O) + (E/L)$	トイレの女性側の出入り口前に停めて	0

表 3: スコアモデルの精度

アクション	S	R-L	R-L\O	R-L/L	R-O/O	R-E\O	R-(O/O)\O	R-(O/O)\L	R-O	R-(L/L)\O	R(L/L)\L	R-E\O
精度	0.910	0.503	0.414	0.042	0.401	0.297	0.360	0.039	0.750	0.080	0.001	0.178
再現率	1	0.557	0.853	0.280	0.964	0.991	0.964	0.923	0.911	0.611	0.25	0.971
F 値	0.953	0.528	0.557	0.073	0.566	0.456	0.525	0.075	0.823	0.141	0.001	0.301

カテゴリを含む文を中心にデータを増やす必要性があると考えられる。

5.2 パーサの精度の測定

5.2.1 実験設定

未知語の処理に対する CRF の学習には, CRF Suite[†] を利用した。対象データには, 上記の実験と同じデータを用いた。10-分割交差検定を行い, 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。

表 4: 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	0.592	0.722	0.821
なし	0.603	0.704	0.857

5.2.2 実験結果と考察

精度の結果を表 4 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ものが考えられる。

[†]<http://www.chokkan.org/software/crfsuite/>

6 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 法を用いた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み, より複雑な文を解析できる文法規則の追加や CCGBank の拡充, さらにグラウンディンググラフの構築を目指す。また未知語の処理で予測するカテゴリをランキング形式にし, 上位 n 位までのカテゴリを文法規則に追加できるようにしたい。

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. AAI, 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.