構造を含む生成規則を扱える拡張型チャートパーザ
— Schartパーザ* の実装 —

川辺 諭† 宮崎 正弘‡

†科学技術振興機構 ‡新潟大学大学院自然科学研究科

1 はじめに

CFG 規則の右辺に構造を記述することができる拡張型のチャートパーザ “**Schart パーザ**” を提案する。

従来の構文解析では文法規則の右辺に部分木の構造を記述することができず、個々の部分木は別途文法規則を記述する必要があり、文法数と構文多義数の増加の一因となっていた。この点を解決するために、CFG 規則に部分木構造を埋め込むことができる “**構造化 CFG**” を導入する。構文解析は逐次型のボトムアップチャート法で行なう。構造化 CFG を用いることで文法の記述量が削減され、解析処理における構文多義が抑制される様子を示す。

2 構造化 CFG による文法の記述

従来の構文解析法では文法規則の右辺に部分木の構造を記述することができず、個々の部分木に関しては部分木を生成する文法規則を個別に記述する必要があった。そのために文法の記述量が増え、構文多義が増加する一因となっていた。この点を解消するため、本パーザでは文法の記述形式として、従来の CFG 規則に構造情報を埋め込んだ “構造化 CFG” を導入する。

2.1 構造化 CFG の記述形式

構造化 CFG は LISP 言語の S 式形式で記述する。car 部が CFG 左辺、cdr 部が CFG 右辺に対応し、cdr 部に構造を記述することが可能である（図 1）。

```
■ 従来のCFG
N1->N2,N3,..
■ 構造化CFG
(N1 N2 N3 ..)
■ 構造を埋め込んだ例
(N1 N2 (N3 N4 N5) ..)
```

図 1：構造化 CFG の記述形式

図 2 は、英語文法の第 4 文型の動詞句（動詞、間接目的語、直接目的語）に関して、従来の CFG の記述と構造化 CFG の記述を比較したものである。文法を構造化して記述することで文法数が減り (この例では 3 から 1 に減少)、あらかじめ部分木の統語構造情報が埋め込まれているために、解析時に発生する構文多義の数が抑制されるといった利点がある。

```
■ 従来のCFG（→文法数3）
vp4->v4,iobj,dobj
iobj->np
dobj->np
■ 構造化CFG（→文法数1）
(vp4 v4 (iobj np) (dobj np))
■ 構造化CFG（木構造表示）
|-vp4
  |-v4
  |-iobj
  | |-np
  |-dobj
    |-np
```

図 2：構造化 CFG の例（英語文法動詞句第 4 文型）

2.2 反復記号と選択記号

文法の記述量を削減するために、非終端記号の一定回数の繰り返しを示す**反復記号** “*”、 “+” と、複数の候補の中から非終端記号を選ぶ**選択記号** “[]” を導入する。ここでアスタリスク “*” は直前の非終端記号の 0 回以上の繰り返し、プラス “+” は直前の非終端記号の 1 回以上の繰り返しを表す（図 3）。

*“エスチャートパーザ” と読む。S は構造化／ structured の意。

```
■ 反復記号の例
(N1 N2 N3*)=(N1 N2)
            |(N1 N2 N3)
            |(N1 N2 N3 N3)
            |..
■ 選択記号の例
(N1 [N2 N3])=(N1 N2)
             |(N1 N3)
```

図 3：反復記号と選択記号

2.3 字面指定記号

また、構文多義の発生を抑制するために、CFG 規則中で終端記号を指定する**字面指定記号** “:” を導入する（図 4）。

```
■ 英語to不定詞句
 (inf prep:to vp)
■ 英語to不定詞句（木構造表示）
 |-inf
   |-prep:to
   |-vp
```

図 4：字面指定記号

3 Schart パーザの実装

Schart パーザによる構文解析は、逐次型のボトムアップチャート法で行う。構文解析のツールとしてはこれまでに一般化 LR 法（富田法）を用いたパーザ [1] [2] などが提案されている。一般化 LR 法は解析途中の状態を計算しておくことであらかじめ準備された状態遷移表にそって決定的な解析を行うために高速であるといった利点があるが、解析途中の状態が人間にとって直感的に把握しにくく、文法が実装しづらいといった難点がある。ボトムアップチャート法は、一般化 LR 法と比較すると余分な多義を発生する可能性はあるが、解析途中の状態が直感的に分かりやすく、文法を実装しやすいといった利点がある。

3.1 文法リーダと文法辞書

文法は構造化 CFG の部分木データを最小単位とし、それらがリンクされた木構造として読み込まれる。読み込まれた文法は文法辞書に格納される。文法辞書は検索処理の高速化をはかるため、木構造左隅の品詞をキーとするハッシュとして実装している。

3.2 構造化 CFG を用いた解析弧

構文解析部では、従来のチャート法と同様に、解析中の部分木を解析弧（弧またはエッジ）と呼ばれる構造体に保存する。解析弧を e、文中の位置を i、解析位置をドット・、とおくと、解析弧は図 5 の形式で示される。

```
e = <i, ドット・が挿入された構造化CFG>
```

図 5：解析弧

また従来のチャート法の用語を踏襲して、ドットが構造化 CFG 規則の右端まで進んだ解析済の弧を “不活性弧”、ドットが規則右端にたどりついていない解析中の弧を “活性弧” と呼ぶ。

図 6 は解析処理時に生成された解析弧の様子である。

```
■ 構造化CFG
 (N1 N2 (N3 N4 N5) ..)
■ 解析弧の例
 <0, (N1 N2:t1 (N3 (N4 N6:t2)・ N5) ..)>
■ 解析弧の木構造表示
 |-N1
   |-N2:t1
   |-N3
   | |-N4
   | | |-N6:t2
   | |-・     ←解析位置を示すドット
   | |-N5
   |-..
```

図 6：構造化 CFG から生成された解析弧の例

構造化 CFG 右辺の木構造の葉 N2、N4 のそれぞれに関して終端記号 t1、非終端記号 N4 を根とする不活性弧 (N4 N6:t2) が適用され、解析が N4 と N5 の間まで進んでいることをドット・を用いて表現している。

解析弧が完成するまでは、適用された終端記号 t1 や部分木 N4 は保存しておく。解析弧が完成して不活性弧となったときに、構造化 CFG の木構造情報に応じて、解析結果である統語木構造を生成する。

3.3 構文解析部

Schart パーザにおけるパージング処理は、構造化 CFG の統語構造情報が部分木の構造に反映される点

をのぞいて、従来のボトムアットチャート法と同等である。すなわち文中の全単語に関して、図 8 の手順によって解析弧を生成する。解析弧は解析弧倉庫に図 7 の形式で保存される。

```
<文頭からの位置i，次に適用可能な品詞p，解析弧>
```

図 7：解析弧倉庫の保存形式

現在の実装では文頭からの位置を添字とした配列を準備し、配列の各要素として品詞をキーとしたハッシュを格納し、ハッシュの値として複数の解析弧をのリスト形式で保存している。

パージングアルゴリズムは図 8 の通り。

```
procedure Proc_メイン
  for 文頭から文末までの単語wに関して do
    for 左隅に品詞pを持つ全文法gに関して do
      左隅にwを適用した解析弧eを生成する [1]
      Proc_弧処理(e)
    end;
  end;
end;

procedure Proc_弧処理(弧:e)
  if eが不活性弧
    不活性弧処理
  else
    活性弧処理
  end;
end;

procedure Proc_不活性弧処理(弧:e1)
  for 解析弧倉庫中で弧e1と同じ開始位置iと
      適用可能品詞pを持つ活性弧e2に関して
    e2のコピーe3を生成
    e3の次の葉としてe1を適用 [2]
    Proc_弧処理(弧:e3)
  end;
end;

procedure Proc_活性弧処理(弧:e)
  解析弧倉庫に弧eを保存
end;
```

図 8：解析の手順（ボトムアップチャート法）

解析結果となる統語木構造の実体は、図 8 中の [1]、[2] のタイミングで生成する。

3.4 解析例 1 —英語名詞句—

図 9 は構造化 CFG で記述された英語名詞句の文法と、Schart パーザによる統語解析の例である。

```
■ 英語名詞句の例
(np det* adjp* n+ [relp pp*])
(adjp adv* adj)
(pp prep np)
▲ "the pretty doll on the desk" の解析例
(np (det the) (adjp (adj pretty)) (n doll)
 (pp (prep on) (np (det the) (n desk))))
|-np
  |-det
  | |-the
  |-adjp
  | |-adj
  |   |-pretty
  |-n
  | |-doll
  |-pp
    |-prep
    | |-on
    |-np
      |-det
      | |-the
      |-n
        |-desk
```

図 9：英語名詞句の解析例（新）

図 10 は従来の CFG で記述された英語名詞句の文法と、統語解析の例である。

```
■ （参考）従来のCFGによる英語名詞句文法の例
np->np0
np->np0,pp
np0->det,np1
np0->np1
np1->adjp,n
np1->n
pp->prep,np
▲ （参考）生成される木構造の例
(np (np0 (det the) (np1 (adjp (adj pretty))
 (n doll))) (pp (prep on) (np (np0 (det the)
 (np1 (n desk))))))
 |-np
   |-np0
   | |-det
   | | |-the
   | |-np1
   |   |-adjp
   |   | |-adj
   |   |   |-pretty
   |   |-n
   |     |-doll
   |-pp
     |-prep
     | |-on
     |-np
       |-np0
         |-det
         | |-the
```

```
|-np1
  |-n
    |-desk
```

図 10：英語名詞句の解析例（旧）

図 9、図 10 の解析結果を比較すると、構造化 CFG を用いた解析は反復記号*などを用いて簡略化した記述ができるために、文法の記述量が少なく可読性が高い。また、文法を分割して記述する必要がないため、図 10 に見受けられる np0、np1 のような不要な木構造ノードが発生しないといった利点がある。

3.5 解析例 2 —英語動詞句—

図 11 は構造化 CFG で記述された英語第 4 文型動詞句の文法と、Schart パーザによる解析結果である。

```
■ 英語動詞句の例
 (vp4 v4 (iobj np) (dobj np))
▲ "gave my son some money" の解析例
(vp4 (v4 gave) (iobj (np (det my) (n son)))
  (dobj (np (det some) (n money))))
|-vp4
  |-v4
  | |-gave
  |-iobj
  | |-np
  |   |-det
  |   | |-my
  |   |-n
  |     |-son
  |-dobj
    |-np
      |-det
      | |-some
      |-n
        |-money
```

図 11：英語動詞句の解析例

構造化 CFG では間接目的語 iobj や直接目的語 dobj などといった木構造上の中間ノードを、文法に直接埋め込むことができるため、記述量を削減することができる。さらに、文法にあらかじめ統語構造の情報が与えられているため、解析処理の負荷が軽減される。

また、英語のように特定の単語（動詞）によって統語構造が大きく左右される言語では、それらの単語がキー（部分木の左隅）となるように文法を記述することによって、不要な解析弧の生成を抑制することができる。この例の場合は解析弧 vp4 は動詞 gave の品詞 v4 をキーとして生成されており、その他の不要な（動詞 give が取り得ない文型の）動詞句に関しては、解析弧が生成されない。

4 おわりに

構造を含む生成規則を扱える拡張型のチャートパーザを試作した。部分木に関する生成規則を個別に準備せずに大域的な木構造に埋め込んでおく構造化 CFG を導入することで、文法の記述量を削減し、構文多義を抑制できることを示した。

今後の課題としては

- 構造化 CFG と従来の CFG の記述能力の比較
- Schart パーザによる実用レベルの英語／日本語／中国語文法の実装
- Schart パーザの定性・定量評価
- 文法適用時の制約条件となる補強項機能の実装

などがある。

参考文献

[1] 沼崎, 田中：SGLR：逐次型一般化 LR パーザの Prolog による実現, 情報処理学会論文誌, vol.32, No.3, pp.396–403(1991).

[2] 五百川, 宮崎：痕跡処理のための逐次型一般化 LR パーザ SGLR の拡張, 言語処理学会第 4 年次発表論文集, pp.314–317(1998).

[3] 田中：自然言語解析の基礎, 産業図書 (1989).

[4] Tomita：Generalized LR Parsing, Kluwer Academic Publishers(1991).

[5] Bunt, Tomita： Recent Advances in Parsing Technology, Kluwer Academic Publishers(1996).