

形態素意味解析システム LAX による語の意味構成

2F-4

久保幸弘

(財) 新世代コンピュータ技術開発機構

1 はじめに

ICOT 第2研究室では、自然言語処理システムの構築に必要な文解析・文生成などのモジュールの開発環境を LTB(Language Tool Box)としてまとめている。形態素意味解析システム LAX は、この LTB のモジュールの一つであり、構文意味解析システム SAX の前処理部に相当する。従来の LAX[1]では、べた書きの日本語文から語(文節)の並びを切り出すいわゆる形態素解析の機能とその開発環境を提供してきた [3]。

今回 LAX に、形態素解析結果を元に語の構文情報や意味情報を構成する機能を追加したので報告する。従って LAX での処理は単なる形態素解析ではなく、語の構文情報や意味情報も解析するため、これを形態素意味解析と呼ぶことにする。この機能により、構文解析モジュールで必要な語の構文情報や意味情報を LAX で容易に構成できるようになった。なお、LAX の意味構成部の開発環境に付いては [4]で報告する。

2 形態素意味解析

SAX[2]などの DCG をベースとする日本語構文解析システムでは、前処理としてべた書きの文を語の系列に変換する形態素解析が必要になる。従来の LAX を用いればこれは可能であるが、この方法では構文ルール中に LAX で出力される各語の構文意味辞書を持つ必要がある。

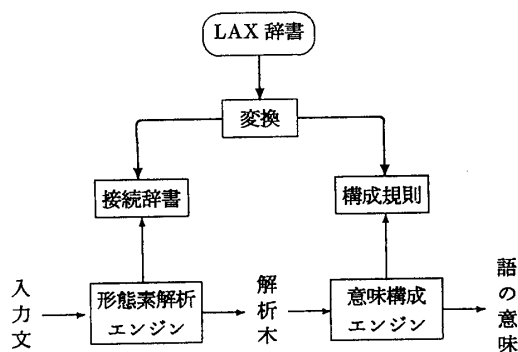


図 1: 形態素意味解析の流れ

Construction of Semantic Structures of Words
in terms of the Morphological and Semantic Analyzer LAX
Yukihiro KUBO, ICOT

Id : [[(Sur, Tok, Cat, Nid, Stt, Fea, Sno), ...], ...]

Id : 語の開始文字位置	Nid : 後接する形態素の 開始文字位置
Sur : 形態素の表層	Stt : 状態遷移表名
Tok : 形態素のトークン	Fea : 接続素性名
Cat : 形態素のカテゴリ	Sno : 意味構成規則番号

図 2: 形態素解析結果の一般形

したがって語の数が增加するにつれて構文ルール数も増え、構文文法の開発に専念できないし、辞書を2箇所を持つのはシステムの開発効率を低下させる。そこで LAX に、語の構文情報や意味情報などを構成する機能を新たに導入した。これにより、SAX では日本語の語の間の構文構造を解析するだけで良い。図 1 に LAX の形態素意味解析の流れを示す。

また助動詞の語順のように、3型文法である形態素の接続規則だけでは記述しきれない、あるいは記述できても規則が非常に複雑になるようなものが、意味構成規則として簡単に記述できるため、より厳密な形態素解析処理が可能となる。意味構成規則を用いることにより無意味な語の候補を、意味情報を用いて効果的に落とす処理もおこなうことができる。

さらに、意味構成の構文解析部への出力は語を構成する形態素のどこでもおこなえるため、そのタイミングを変えることにより係り受け文法 [5] 用の文節の構文意味構造を構成したり、句構造文法用の語の意味構造を構成することができる。

3 意味構成法

LAX による語の意味の構成は、形態素解析がすべて終了した後におこなう。これは、形態素解析レベルでは一般に語にまとまる前に落とされる候補がかなりあり、そのような候補に対する無駄な意味の構成を避けるためである。

LAX が出力する形態素解析結果には語を形成する各形態素に、図 2 のように、形態素の表層とトークンおよびカテゴリ、後接する形態素の開始文字位置、接続時に使用した状態遷移表名と接続素性名、意味構成規則の番号が記される。意味構成規則は、形態素辞書に後述する記述形式に従って記述する。

語の意味構成処理はこの形態素解析結果に現れる語について、その語を構成する形態素の持つ意味構成規則を順に適用することによりおこなう。ある形態素の意味構

```

意味構成規則 ::= ' [ 変数宣言 ] '
                | ' [ 変数宣言 ] , 構成規則並び '
変数宣言      ::= ' [ { 入力宣言 } , 出力宣言 ] '
入力宣言      ::= ' no ( 変数 ) ' | ' in ( 変数 ) ' | ' sur ( 変数 ) '
                | ' morph ( 変数 ) , 変数 '
出力宣言      ::= ' out ( 変数 ) ' | ' sax ( 変数 ) '
構成規則並び ::= 構成規則 { , 構成規則 }
構成規則      ::= データ型操作呼出し | 条件文
条件文        ::= ' cond ( 条件対並び { ' , [ 構成規則並び ] ' } ) '
条件対並び   ::= 条件対 { ' , 条件対 }
条件対        ::= ' [ 素性指定 { ' , 素性指定 } , [ 構成規則並び ] ' ] '
素性指定     ::= 素性名 | ' prev ( 素性名 ) ' | ' next ( 素性名 ) '
素性名       ::= 状態遷移表名 ' ( [ 接続素性名並び ] ) '
接続素性並び ::= 接続素性 { , 接続素性 }

```

図 3: 意味構成規則記述形式

成規則が適用される際には、その規則への入力データとして、直前の形態素までに構成された意味構造、形態素の表層とトークンおよびカテゴリ、後接形態素との接続に使用した状態遷移表名と接続素性名が渡される。

出力データは、後接形態素へ入力意味構造として送ることができ、語の構成途中であってもその形態素までの意味構造を語の意味構造として構文解析部に送ることができる。語の先頭の形態素への入力データは未定義変数である。

4 意味構成規則の記述形式

従来の LAX 辞書では、各形態素に対して自分自身の識別子である左方接続素性と、後接しうる形態素の識別子の集合である右方接続素性を記述する。ここに更に図 3 で定義される意味構成規則をつけ加える。意味構成規則は変数宣言部と構成規則の並びとからなる。変数宣言部では、意味構成エンジンと構成規則とのあいだでデータのやりとりをおこなうための変数を定義する。出力変数宣言された変数を除くすべての変数は、構成規則が呼ばれる時点で意味構成エンジンにより値の束縛を受ける。出力変数には構成規則が構成した意味構造を束縛する。宣言できる変数とその意味は次の通りである。

no(No)	: 語の識別番号 (文中で一意)
in(In)	: 前接形態素の出力データ
sur(Sur)	: 形態素の表層
morph(Tok, Cat)	: 形態素のトークンとカテゴリ
out(Out)	: 後接形態素の入力データ
sax(Sax)	: 語の意味構造を出力

意味の構成規則は、後述する語の意味構造に対する操作命令 (意味データ型操作) か条件文からなる。条件文により、前後の形態素との接続状況に応じて意味構造に対する操作を自由に変えることができる。この記述形式で書かれた LAX 辞書の例を図 4 に掲げておく。

この記述形式で書かれた意味構成規則は、トランスレータにより意味構成プログラム・モジュールに変換され、意味構成エンジンにより随時呼び出される。

5 語の意味構造

LAX で構成される語の意味構造は、属性と値の対で表されるレコード構造である。その構造とそれに対する操

```

が :: end( [ 文節 ] )
    && [ 句読点 ( [ 読点 ] ) ,
        end( [ 文節 ] ) ]
    $$ [ [ in( SEM ) , sur( Sur ) , morph( Tok , Cat ) , out( SEM ) ] ,
        role( SEM , [ moding , case ] , Tok ) ,
        cond( { [ end( [ 文節 ] ) ] ,
                [ role( SEM , [ sur ] , Sur ) ] } ) ,
        cond( { [ end( [ 文節 ] ) , 句読点 ( [ 読点 ] ) ] ,
                [ role( SEM , [ moded , bar ] , 1 ) ,
                  role( SEM , [ moding , cat ] , 適用詞 ) ] } )
        ] ] .

```

図 4: 形態素意味辞書の記述例

作命令は予めユーザが抽象データ型として定義しておく。意味の構成規則部にはこのデータ型の操作命令しか許さないことにより、意味構成プログラムのモジュラリティを良くしている。なおこのデータ構造とその操作命令は、構文意味解析システム SAX でも使用される。意味構造のテンプレートを与えると、属性値へのアクセスなどの基本的な命令はトランスレータにより自動的に生成される。ユーザは、この生成されたプログラムに意味構成に必要な命令を追加して意味構成規則中で使用する。

6 おわりに

形態素に記述された意味構成規則を用いて、語の意味を構成する機能を LAX に付加した。形態素の接続状況に応じて意味構成処理を変えられるため、構文解析部に必要な構文情報や意味情報を容易に構成できるようになった。また、接続規則だけでは記述が困難であった形態素の順序規則などの記述や、意味情報を用いた形態素解析の曖昧さの解消が可能となった。

今後の課題は、意味構成規則の記述力の強化と、係り受け文法と句構造文法に対する意味構成規則の記述実験を進めること、さらに文生成部の形態素処理部との辞書データの共用があげられる。

参考文献

- [1] 杉村領一, 赤坂宏二, 久保幸弘, 松本裕治, 佐野洋, 論理型形態素解析 LAX, *Proceedings of the Logic Programming Conference '88*, pp213-222, 1988
- [2] Y. Matsumoto and R. Sugimura, A parsing system based on Logic Programming, *Proceedings of the International Joint Conference of Artificial Intelligence*, 1987
- [3] 久保幸弘, 妙泉正隆, 佐野洋, 赤坂宏二, 杉村領一, LTB-形態素解析システム LAX の開発環境, 情報処理学会第 37 回 (昭和 63 年後期) 全国大会論文集 (II), pp1078-1079, 1988
- [4] 米田順美, 久保幸弘, 白石智子, 妙泉正隆, LAX インタプリタと意味構成部のデバッグ環境の実現, 情報処理学会第 39 回 (平成元年度後期) 全国大会論文集
- [5] 福本文代, 杉村領一, 構文意味解析システム SAX における係り受け解析 - 構文意味情報の構成について -, 情報処理学会第 39 回 (平成元年度後期) 全国大会論文集