語彙化されたツリーオートマトンに基づく会話文翻訳システム

山端 潔 安藤 真一 三村 清美
NEC C&C メディア研究所
e-mail: {yamabana, ando, mimura}@ccm.cl.nec.co.jp

1. はじめに

メディア処理技術の進歩とともに、自然言語処理の対象が多様化している。二者間の対話にあらわれる会話文はその一例であり、音声言語処理技術の進歩とともに、言語処理の対象としての重要性を大きく増しつつある。

我々は、このような会話文を対象とした機械翻訳システムを開発している。そのために、汎用の一般文法と、単語やドメインに特有の個別文法を統一的に記述するのに適した新しい文法の枠組みとして、語彙化ツリーオートマトンに基づく文法記述形式を開発した。本稿では、この文法形式について述べるとともに、この形式に基づく英日会話文翻訳モジュールの実装について報告する。

2. 会話文の処理と文法形式

会話文には、書き言葉の文法を逸脱した会話特有の表現や、特定の単語の組み合わせに基づく熟語的表現など、多様な表現が現れる。そのため、翻訳においては、汎用的な一般文法による処理に加えて、単語毎の個別処理を詳細に記述する必要がある。前者はルール的、後者は事例的な記述となるため、これらを統一的に扱う枠組みが重要となる。

ルール的な文法形式をベースとして、単語に依存したパタン的な文法規則を導入する方法として、語彙項目の導入により文法規則を特殊化するアプローチがある。例えば、武田[8]は、書き換え規則の右辺にヘッドの単語を指定するヘッド制約を導入することにより、文脈自由文法の枠内で、表層レベルのパタン的規則を導入している。また、長瀬ら[4]は、一般ルールと用例パタンを組み合わせて用いる方式を示している。

一般に、すべての文法規則が単語に関連付けられ、特殊化された文法形式に、語彙化文法の枠組み[5]がある。語彙化文法は、一般規則に対する語彙的影響を捉えることを目的の一つとして導入されたものだが、単語に依存した特殊規則の記述が容易であることは、その定義からも明らかである。従って、一般テキストに対するカバレッジを確保しつつ、会話文に出現する特殊かつ多様な表現に対応するためには、語彙化文法は採用すべき文法形式の有力な選択肢となる。

語彙化した句構造文法[1]の構成要素は、ツリー、ツリー間の演算と、ツリーを単語に関連付ける方法の3つである。例えば LTAG (Lexicalized Tree Adjoining Grammar) [8] では、ツリー間の演算として、ツリーの連結に加えて、auxiliary tree による adjunction 演算を持ち、すべてのツリーに語彙項目を持たせる形で単語への関連付けを行う。別の例として、ヘッドの概念を持つ CFG の場合、前述のように、各規則にヘッド単語を指定する制約を設けて語彙化することが可能である。

ところが、既存の語彙化句構造文法形式には、いくつか課題がある。

まず、語彙化されているのはツリーのみで、ツリー間の組み合わせを制御するツリー演算は語彙化できない。単語に特有の文法規則を記述する際には、一つのツリーだけでなく、複数のツリーの組み合わせ方やその適用条件等、規則間の制御を個別に記述したい場合がしばしばある。しかし、従来の語彙化文法の枠組みでは、ツリー演算は必ずグローバルな定義を持つため、このような目的には、例えばノードの品詞と素性構造を通じて間接的に制御するしかない。

第二に、例えば LTAG もそうだが、語彙化の代償として、しばしば連結以外のツリー演算が必要となり、構文解析アルゴリズムを複雑化する。ツリー中に単語を要求する形で語彙化を定義した場合、連結演算だけでは、ルートから語彙項目までの距離が一定値以下のツリーしか作れない。この制約を、例えば LTAG では、ツリー中に別のツリーを挿入する adjunction 演算が補償している。別の言い方をすれば、単語が支配を及ぼす範囲である EDOL (Extended Domain of Locality) をツリーの形で明示的に表現するためには、連結以外のツリー演算と、専用の解析アルゴリズムが必要となる。

次節では、これらの課題に対処するために、ある単語をヘッドとするツリーを、その単語に関連付けたツリーオートマトンとして表現することを特徴とする語彙化ツリーオートマトン文法を導入する。

3. 語彙化ツリーオートマトン文法

一般に、句構造文法は、非終端記号と終端記号(単語)をノードに持つ有限のツリーの集合と、ツリーを組み合わせて別のツリーを構築するツリー演算の組として定義される。例えば、文脈自由文法は、高さ1のツリーの集合と、非終端記号の一致によるツリーの連結演算により定義される句構造文法である。また、語彙化文法は、各ツリーに単語が関連付けられた文法として定義される[2]。

本稿で提案する語彙化ツリーオートマトン (Lexicalized Tree Automata, LTA) 文法の基本的な

[1] 本稿では、文法形式として、句構造文法のみを考えるが、提案する形式を依存文法等他の形式に拡張することは容易である。

[2] よく採用される定義については[8]を参照されたい。

アイディアは、単語に付随しEDOLを表現するツリーの集合を、直接的なツリーの列挙ではなく、ツリーを受理するツリーオートマトンにより間接的に定義することにある。

3.1. 文法の定義

語彙化ツリーオートマトン文法(LTA文法)とは、各単語に対し、その単語をヘッドワードとする要素ツリーの集合と、要素ツリーを指定点で連結して構成されるツリーのうち、文法が許容するツリーのみを受理するツリーオートマトンの二つが付随する文法形式である。受理されたツリーの集合は、その単語をヘッドとして成長可能なツリーをあらわす。終状態に達すると、ツリーのルートに非終端記号が与えられる。一方、各単語から成長した(i.e. その単語のツリーオートマトンが受理した)ツリー同士の演算は、非終端記号の一致による通常の連結演算とする。

すなわち、LTA 文法とは、ツリー演算を、ある単語をヘッドとするツリーを形成する部分(ローカル文法)と、それらのツリーを連結する部分(グローバル文法)に分解し、前者を単語に付随するツリーオートマトンとして表現した文法形式である。以下に形式的な定義を示す。

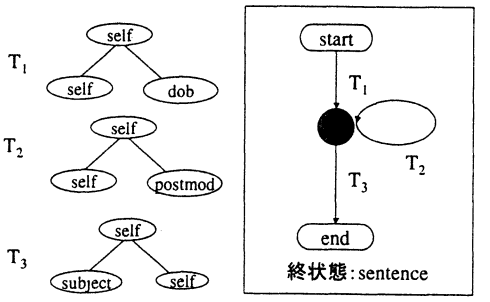
定義(語彙化ツリーオートマトン文法):

S を終端記号(単語)の集合、NT を非終端記号の集合とする。E_wをSの要素wに関連付けられたツリーの集合とする。E_wに属するツリーは、SまたはNTの記号をノードに持ち、かつ、リーフの一つおよびルートが NT の特殊記号 self によりマークされているものとする。A_wは、wに関連付けられたツリーオートマトンであり、E_w のツリーをselfのノードで連結してできるツリーからなる集合の、ある部分集合T_wを受理する。N_wは、A_wがツリーを受理したときに、ルートノードに与えられる非終端記号の集合である。各単語wに対し、3 つ組(T_w, A_w, N_w)は、A_wによりwに関連付けられたローカルツリーをあらわす。

語彙化ツリーオートマトン文法 G とは、$\cup T_w$に属するツリーを基本ツリーとし、これらのツリーの間の連結演算を基本演算とするツリー文法である。■

3.2. 注意

- LTA 文法では、ある単語に付随するツリーの集合 T_w は一般には無限集合になりうる。そのため、厳密には、LTAG 等で一般的な語彙化文法の定義(各単語に有限のツリーが付随する)から、はずれることがある。しかし、ツリー集合は、有限な記述を持つツリーオートマトン A_wで生成される点で有限性を持つため、この逸脱は本質的ではない。
- LTA 文法におけるオートマトンは、ツリーの集合 T_wを受理するように構成されている点で、広い意味でのツリーオートマトンである。一方、上述の定義では、オートマトン A_wを、構成要素たるツリー(E_wの元)をアルファベットとする文字列を受理するストリングオートマトンとして定義した。一般に、ある単語をヘッドとするツリーは、その単語からルートに至る"背骨"(spine)を軸に、構成要素のツリーを並べることにより、構成要素ツリーのツリー列と同一視することができる。この同一視により、ツリーオートマトンと、ツリー列を受理するストリングオートマトンを同一視することができる。本稿では、これらの二つの表現は、状況に応じて適宜使い分けるが、同じものであることに注意されたい。
- 上述の定義では、一単語に一つのオートマトンしか関連付けていないが、曖昧性を許すように拡張することは容易である。



(a) ツリー集合　(b) ツリーオートマトン

図1:eatの辞書内容

3.3. 例

図 1 は、動詞 eat の持つツリー集合とツリーオートマトンの一例である。eat からは、自身をヘッドとして、図 2 に示すツリーが成長するものとする。eat に付随するツリー集合(図 1(a))は、直接目的語を取り込むツリーT_1、副詞等の自由修飾要素を取り込むツリーT_2、および主語を取り込むツリーT_3 からなる。ルートおよび葉の一箇所がself とマークされているのは、このノードを重ね合わせながら図 1(b) のオートマトンによるツリーの受理が進むことをあらわす。

始状態にあるオートマトンは、まず T_1 を受理する。これは、図 2 で、eat が dob を取り込んで高さ 1 のツリーを作ることに対応する。次に、オートマトンは、T_2を 0 回以上、任意回受理する。これは、図2で、postmod とマークされたノードを任意回取り込んでツリーが成長する部分に対応する。最後にオートマトンはT_3を受理して終状態に至る。これは、図 2 で、ツリーが subject を取り込んで、eat をヘッドとするツリーが完成したことをあらわす。出来上がったツリーには、非終端記号 sentence が与えられる。このように、図 1 の記述と図2のツリーは等価となる。

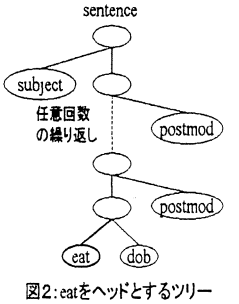


図2:eatをヘッドとするツリー

図3は、文脈依存言

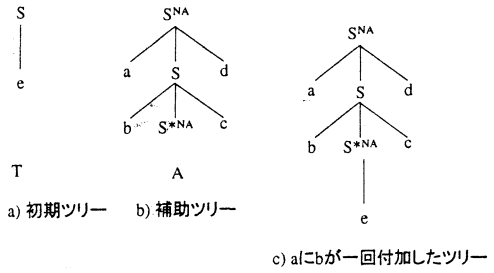


図 3: $a^n b^n e c^n d^n$ を生成するLTAG

語 $a^n b^n e c^n d^n$ を生成する LTAG の例であり、図4は、これと同じ言語を生成する LTA 文法の例である。図4では、e に付随するツリーオートマトンは、$(T_2)^n(T_1)^n$ を受理するプッシュダウンオートマトンである。一般に、任意の LTAG に対し、プッシュダウンオートマトンを用いて弱同値な LTA 文法を構成することができる。また、文脈自由文法と等価な LTA 文法は、正規ストリングオートマトン(正規ツリーオートマトン)の範囲内で構成できる。このように、オートマトンのクラスを変えることにより、LTA 文法の枠組みで様々なクラスの文法が受理できる。

4. 構文解析アルゴリズム

一般の LTA 文法に対し、ボトムアップチャート法をベースとした構文解析アルゴリズムが定義できる。CFG の場合、アクティブエッジは、ルール右辺の受理がどこまで進んだかを示すドットつきルールにより表現されるが、LTA 文法では、アクティブエッジは、ローカルツリーの受理がどこまで進んだかを示すツリーオートマトンの状態として表現する。その他の点では、解析アルゴリズムは CFG と同様である。

解析の効率化には、エッジのパックが重要であるが、その可否は、オートマトンの未適用部分および非終端記号の一致で判定する。なお、解析途中での枝刈りや部分解の利用も、CFG の場合と同様に行える。

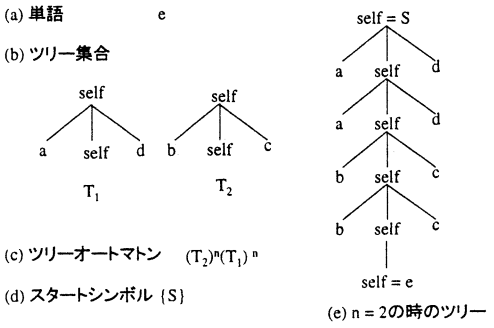


図4: $a^n b^n e c^n d^n$ を生成するLTA文法

5. 翻訳システムの構築

LTA 文法の枠組みに基づき、旅行会話文をターゲットとした英語・日本語間の双方向翻訳システムを構築した。本節では、開発した翻訳エンジンおよび英日方向の翻訳文法・辞書について述べる。

5.1. 翻訳エンジン

LTA 文法では、すべてのツリーに加えてツリーオートマトンが語彙化され、単語辞書内に分散記述されている。従って、その実装にあたっては、これらを効率的に共有する仕組みが重要である。そのために、ルールテンプレート機構と共通ルール機構を設けた。ルールテンプレートは、ツリー集合とツリーオートマトンの組を単語間で共有する仕組みであり、初期チャート作成直後に実体がロードされる。共通ルール機構は、個々のツリーをツリー集合間で共有する仕組みである。共有されたツリーはポインタとして表現され、実際の参照時に実体がロードされる。

言語変換には同期導出[6]をベースとする方式を採用した。ただし、変換後のツリーを、構文木自体ではなく、構文木を生成する手続きの呼び出し関係の表現とすることにより、生成過程の自在な制御を可能としている。

5.2. 英日翻訳文法と辞書

英語文法は、標準的なXバー理論に準拠した句構造を採用し、素性構造により補強している。また、個々のツリーに、ツリーの受理時に実行される補強項を設け、解析過程の詳細な制御を可能にしている。

英日辞書は見出し語数約 7 万である。そのうち、個別のツリーセットとツリーオートマトンの記述が必要となった単語は数千であった。

LTA 文法では、異なる単語をヘッドに持つツリー間の連結には、従来と同様、品詞(素性構造)のマッチングを用いているが、この際に語彙的な統語的能力の違いを細かく表現する手段として、Link Grammar[7]と同様、品詞名ではなく文法関係名を中心に記述するアプローチをとった。

LTA 文法では、単語に付随するローカル文法がツリーオートマトンとして表現されており、オートマトンの合成演算の概念が導入可能である。実際、今回の実装でも、異なる単語をヘッドとするツリーの連結において、子供となるツリーが親となるツリーのオートマトンに別のオートマトンを連結・挿入する機能を実装し、文法記述に利用した。これにより、ヘッド以外の単語がヘッドの単語のローカル文法に修正を与える状況を表現している。

5.3. 実装と予備的評価

英日翻訳部は、C++で実装され、Windows NT 上で動作する。現在、Pentium II 300MHz 以上の CPU、60MB 以上のメモリを必要とする。ただし、メモリの大部分は各種のスタティックデータであり、ツリー集合やオートマトンは効率的に共有されている。

海外旅行の場面で見られる種類の会話文を対象に、予備的な訳質評価を行った。約4万文の中から500文をランダムに選択し、本システムと既存の商用翻訳システム(辞書強化済み)でそれぞれ翻訳した。この結果を、システム名を伏せてランダムに混ぜ、4段階(natural、good、understandable、bad)に分類した。その結果、natural に分類されたものが、既存のシステムに対して約45%増加する一方、bad に分類されたものは約40%減少し、本システムが海外旅行会話のドメインで有効であることを確認した。

6. 考察

LTA 文法は語彙化文法の一形式であり、文法の実効的な大きさが、入力文中の単語に関連する部分に縮小される等の、一般の語彙化文法の利点[5]を共有する。

文法の能力としては、ツリーオートマトンのクラスを(ツリー列に対するストリングオートマトンとしての表現で)正規オートマトンに限ると、その生成能力は文脈自由文法と等価である。また、プッシュダウンオートマトンを使うと、LTAG と弱同値な文法を構成することができる。一歩進んで、ツリーオートマトンを、(ツリー列を受理する表現において)Tree Adjoining Language を受理するように構成した場合にどのようなクラスの言語が受理できるかは興味深いところである。

ツリーオートマトンは、EDOL の有限な表現となっている。EDOL の非有限性は、オートマトンが生成するツリーの非有限性として表現されており、adjunction のような特殊なツリー演算が不要なため、オートマトンの具体形によらない統一的な構文解析アルゴリズムが存在するのも利点である。

従来、話し言葉の翻訳を目的として、構成素境界を手がかりに表層レベルのパタン文法を記述する手法が提案されている[3]。この手法では、文法規則は比較的単純なパタン規則であり、記述が容易である。しかし、抽象度の高い汎用規則を記述したり、詳細な構文意味規則を記述するには不向きであると思われる。一方、提案する手法によれば、抽象度の高い一般規則から具体性の高い個別パタンまでを、統一的な枠組みで記述・管理できる利点がある。また、語彙化 CFG[4]に基づく手法と比較しても、規則の適用を単語ごとに直接的かつ精密に制御できるという特徴があり、この特徴は、実際の文法記述においても訳質向上に寄与している。文法としてオートマトンを記述するのは一見複雑に思えるかもしれないが、実質は単語をヘッドとするツリーを定義する従来の文法記述の作業と同等であり、かえって、ツリーがうまく成長するように品詞や素性構造を操作する必要がないぶん、見通しの良い文法記述が可能となっている。

本稿の手法と、オートマトンベースの解析の手法として従来提案されている手法[1,2]の相違は、従来の手法のオートマトンが、非終端記号列を受理する有限オートマトンであるのに対し、LTA 文法ではツリーを受理するツリーオートマトンである点にある。これは、受理可能な言語のクラスに影響を与える。また、本手法では、単語に関連付けられたツリーの作るローカル文法の概念が明確化されているのも特徴である。すなわち、同じ単語をヘッドとするツリーの成長は、その単語のローカル文法として、単語に関連付けられたツリーオートマトンにより表現されている。この部分を様々に変えることにより、例えば今まで文脈自由文法の枠内で記述してきた文法に、文脈依存性を部分的に導入する(例えば $a^n b^n e c^n d^n$ を導く単語 e を新規導入する)ことも可能である。一方、従来の語彙化文法の枠組みでは、可能なツリー演算は文法定義の一部にビルトインされているため、このような柔軟な変更は困難である。また、[2]においては、オートマトン化は構文解析の高速実装手法として定式化されており、文法形式の能力に影響を与えることはできない。

7. おわりに

語彙化文法の一形式として、語彙化ツリーオートマトン文法の枠組みを考案し、これに基づいて旅行会話文をターゲットとした日英双方向の翻訳システムを構築した。提案する文法形式は、ツリー自体に加えて、同じ単語をヘッドとするツリーの成長を記述するローカル文法を語彙化し、単語に付随するツリーオートマトンとして表現するのが特徴である。これにより、文法規則の適用を単語ごとにきめ細かに制御することが可能となり、抽象度の高い一般規則から、語彙依存の個別規則までを統一的な枠組みで記述することができる。ツリーオートマトンは、単語の支配領域である EDOL(Extended Domain of Locality)の表現となっている。また、ツリーオートマトンの詳細によらない統一的な構文解析アルゴリズムが存在するのも特徴である。構築した会話文翻訳システムによる英日翻訳結果の予備的評価の結果、既存のシステムに比べて実質的な訳質向上が見られ、本方式の有効性を示す結果となった。

今後は、文法・辞書強化によるさらなる訳質向上と、エンジンの小型化・高速化をはかりたい。また、音声処理との結合も重要なテーマである。

参考文献

[1] H.Alshawi: ``Head Automata and Bilingual Tiling.'' ACL'96, pp.167-176 (1996).

[2] R.Evans et al.: ``A structure-sharing parser for lexicalized grammars.'' ACL'98, pp.372-378 (1998).

[3] 古瀬 他:「構成素境界解析を用いた多言語話し言葉翻訳」自然言語処理, 6 (5), pp.63-91 (1999).

[4] 長瀬 他:「ルールベース翻訳とパターンベース翻訳の融合」言語処理学会第4回年次大会, pp.496-499. (1998).

[5] Y.Schabes et al.: ``Parsing strategies with `lexicalized' grammars.'' COLING'88, pp.578-583 (1988).

[6] S.Shieber et al.: ``Synchronous Tree Adjoining Grammars.'' COLING'90, pp.253--258. (1990).

[7] D.Sleator. et al. ``Parsing English with a Link Grammar''. CMU TR CMU-CS-91-196 (1991).

[8] K.Takeda: "Pattern-Based Machine Translation." COLING'96, pp.1155-1158. (1996).