5J－8

パターンベース翻訳システム：PalmTree *

渡辺 日出雄† 武田浩一‡
日本アイ・ビー・エム株式会社 東京基礎研究所§

1 はじめに

我々はSimTran[4, 5]という用例を利用したトランスファーシステムの研究を行なってきた。SimTranの翻訳パターン（知識）は、ソース言語の解析構造の断片とそれに対応するターゲット言語の構造、および、それらの間のノードの対応関係からなっている。全てのノードが文字列を持つ場合、これは翻訳例と見なすことができる。SimTranでは、この種の翻訳例だけでなく、従来のトランスファー知識も翻訳パターンとして保持し、両者を類似度計算により同一の枠組で扱っている。

このSimTranでの翻訳例と既存の文法的な翻訳知識を統一的に扱うという考え方をベースにして、トランスファーだけでなく翻訳の解析から生成までの全過程を翻訳パターンを用いて同期文法[1]の考え方を基にして一つの枠組で処理しようとしたのが、パターンベース翻訳システムPalmTreeである。以後、このPalmTree[1]に関して簡単に説明する。

2 翻訳パターン

PalmTreeの翻訳パターン[2, 3]は、ソース側言語の文脈自由文法(context-free grammar)規則とそれに対応するターゲット側の文脈自由文法規則のペアとなっている。以下に翻訳パターンの例を示す。

```
Bill Clinton → NOUN
               NOUN ← ビル・クリントン
take:VERB:1 a look at.NP:2 → VP:1
               VP:1 ← NP:2 を 見る:1
large numbers of NP:1 → NP:1
               NP:1 ← かなり多くの NP:1
```

数字は、対応関係を表している。ちょうど、左半分がソース側のルール、右半分がターゲット側のルールになっている。[2]矢印の始点側に相当する部分を右辺、終点側に相当する部分を左辺と呼ぶことにする。ルールを構成する項は、終端記号、終端記号＋品詞、あるいは、品詞のいずれかの形式である。終端記号＋品詞は、終端記号が同じであり、かつ、品詞が同じである場合にだけマッチする。通常、ルール項には、各種の属性や手続きを起動するトリガーなどを記述するが、本稿ではスペースの関係で省略する。

3 翻訳処理

翻訳処理は、文脈自由パーサーと同期導出(synchronous derivation)により実現されている。すなわち、この翻訳パターンのソース側を用いて文脈自由パーサーが解析を行ない、その最中に、使用されているパターンのターゲット側部分から生成構造も同時に作成する。これにより、解析が終った時点では、生成側の構造が出来上がっていることになる。この処理一つで翻訳の主要部分は終りとなる。

4 高速化

PalmTreeでは大量の翻訳パターンを用いるので、文脈自由パーサーにとっては使用するルール数の増大となりパフォーマンスの問題が発生する。そこで、以下のような刈り込み処理を行なっている。

4.1 語彙化規則優先原理

規則の右辺項に終端記号（あるいは単語）を持つものを語彙化規則と呼び、そうでないものを一般規則と呼ぶことにする。語彙化規則優先原理とは、ある区間に語彙化規則と一般規則から作られた不活性アークが競合して存在する場合、一般規則から作られたアークを無効にするというものである。

語彙化規則は、含まれる終端記号の数が多いほど優先されるようにコストが付けられる。

例えば、

He takes a look at a map.

という入力のtakeからmapまでの区間に対して、以下のルールがマッチしているとすると、

(r1) take:VERB a look at NP
(r2) take:VERB a NP at NP
(r3) VERB NP at NP
(r4) VERB NP PREP NP

(r4)はこの処理により使われないことになり、残りの(r1),(r2),(r3)はこのこの順番で優先度が設定されることになる。

---

*A Pattern-based Translation System: PalmTree
†Hideo Watanabe (watanabe@trl.ibm.co.jp)
‡Koichi Takeda (takeda@trl.ibm.co.jp)
§IBM Research, Tokyo Research Laboratory
[1]本システムPalmTreeは、弊社のインターネット向け翻訳ソフトウェア「インターネット翻訳の王様」の翻訳エンジンとして使用されている。
[2]誌面の都合で一つのパターンが2行になっているが、本来1行で記述している。

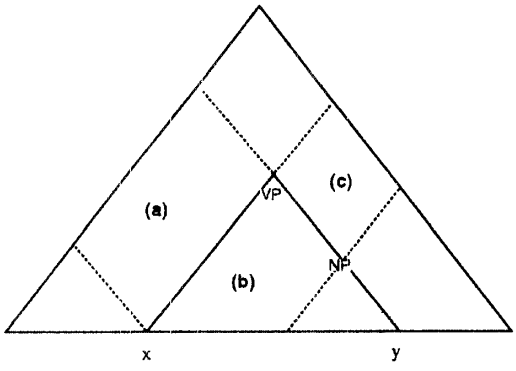
4.2 左隅固定排他規則

ある特殊な表現の場合、その規則がマッチした場合に、それと競合する可能性を排除したいという場合がよくある。この様な場合によく用いられるのが、排他規則である。しかし、通常の排他規則はその規則の左隅と右隅が固定されているため、適用範囲が狭いという欠点がある。そこで、左隅だけが固定されているが、右隅は固定されていない排他規則を処理する仕組みを導入した。

左隅固定排他規則がマッチしている区間を $[x, y]$ とすると、$i < x$ かつ $x < j < y$ である区間 $[i, j]$ と、区間 [x,y] 内の全てのサブ区間（ただし、左隅固定排他ルールの右辺の最右項とマッチしている区間の全てのサブ区間を除く）で

- 排他規則以外の規則の適用を抑止する。
- 排他規則以外の規則から作られたアークを無効にする。

という処理を行なう。



上記の図は、VP ⇐ VERB NP という左隅固定排他ルール[3]がマッチした場合を示しているが、この場合、(a)(b)(c) の区間では、排他規則以外のルールの適用が抑止され、排他規則以外のルールから作られたアークが無効化される。

例えば、以下のようなルールがある。

NP — DET own NP
NOUN — as many as NP
NP — most of NP

この様なルールがマッチすると、上図の (a),(b),(c) に該当する区間での排他規則以外のルールの適用を中止することになる。

5 PalmTree の特徴

従来のトランスファー方式の翻訳システムと比べると、PalmTree は大変簡易なシステムである。解析・トランスファー・生成と３つのモジュールで構成されている部分が単一のモジュールとなっている。このようにシステムの複雑さが低減していることから、PalmTree は以下のような特徴を備えている。

- 翻訳知識の統一的な管理：翻訳知識が翻訳パターンに集約されているので管理が容易である。さらに、一つの知識が、従来のトランスファー方式での解析・トランスファー・生成のそれぞれに効果を及ぼすことができる。
- 容易なチューニング作業：訳質の向上が翻訳パターンの追加により容易にかつ漸進的に行なうことができる。実際、当研究所で開発していたトランスファー方式の翻訳システムに較べてチューニングに必要なコストは１／４に減らすことができた。

6 ユーザーによるパターンの追加

チューニング作業が容易になったとはいえ、エンドユーザーが直接翻訳パターンを記述するのは難しい。しかし、学校で習ったような、簡易なレベルで記述できれば、エンドユーザーでも翻訳パターンの入力が可能である。そこで、エンドユーザーが記述する簡易翻訳パターンのレベルを定義し、これを内部レベルの翻訳パターンに変換する簡易パターンコンパイラを開発した。

例えば、以下のような簡易パターンは、

[PP] between #:1 and #:2 == #:1 と #:2 の 間に

以下のようなパターン表現[4]に変換される。

between NP:1 and NP:2 — PP
PP → NP:1 と NP:2 の 間に

これにより、エンドユーザーは、従来の単語レベルでのカスタマイズに加えて、様々なフレーズレベルでシステムをカスタマイズすることが可能になった。

7 おわりに

パターンベース翻訳システムでは、従来の手法に比べ訳質向上が非常に容易である。基本の文法的翻訳パターン部分は専門の文法開発者が必要であるが、その他の翻訳パターンは、通常の学校で習う程度の英語の知識がある人であれば十分に作成可能である。また、システム構成がシンプルであるため、デバッグが非常にやりやすいという利点もあり、パターン追加の容易さと相まって、チューニング作業を非常にやりやすくしている。

参考文献

[1] Shieber, S., and Shabes, Y., "Synchronous Tree Adjoining Grammars," Proc. of COLING 90, 1990
[2] Takeda, K., "Pattern-Based Context-Free Grammars for Machine Translation," Proc. of 34th ACL, pp. 144–151, 1996
[3] Takeda, K., "Pattern-Based Machine Translation," Proc. of 16th Coling, Vol. 2, pp. 1155–1158, 1996
[4] Watanabe, H., "A Similarity-Driven Transfer System," Proc. of the 14th International Conference of Computational Linguistics, Vol. 2, pp. 770–776, 1992.
[5] Watanabe, H. and Maruyama, H., "A Transfer System Using Example-Based Approach," IEICE Transactions on Information and Systems, Vol. E77-D, No. 2, pp. 247–257, 1994.

[3] 現実にこのルールが左隅固定排他ルールであるわけではない。

[4] 誌面の関係で省略したが、実際にはこれ以外に各種の属性等が付加されている。