

音声・言語日英翻訳システムの変換過程における書き換え規則の制御

1 F - 4

長谷川 敏郎

ATR自動翻訳電話研究所

1. はじめに

素性構造を対象とした書き換えシステムを作成し、これを用いた音声・言語日英翻訳システムの変換モジュールの開発を進めている。本稿では、作成した書き換えシステムのメカニズムと書き換えシステムを用いた変換過程の処理について述べる。

2. 書き換え規則のモジュラリティ

ルールベースの書き換えシステムでは、規則のモジュラリティと、書き換え規則の表現能力が重要な要素になる。変換過程では日英の概念間の直接的な対応関係をとる書き換えだけでなく、表現の間接性に関する書き換えや、イベント的に表現するかオブジェクト的に表現するかといったタイプの書き換え、視点の変換など、種々のタイプの書き換えを必要とする。

ある書き換え対象に対して複数の書き換え可能性が存在する時、一つの規則内で複数の書き換え処理を記述すると、書き換え規則の肥大化、複雑化をまねき、規則の管理を困難にする。また、異なるタイプの書き換えが一つの規則内で記述された時には、その規則が適用されることによって得られる結果を想定することが難しくなり、規則の適用の制御や判断が困難になる。

さらに、規則の独立性はシステムの改良の容易性や拡張性に大きく影響する。規則間の相互依存性が高いと、規則の編集や操作を困難にする。

3. 書き換えシステムにおける規則適用制御

一つの規則を小さく、単純な処理にとどめると、規則の処理内容は明瞭になるが、個々の規則をどのように制御するかという問題が生じる。ここでは、規則の適用を制御するために、書き換え環境と規則適用制約という概念を導入した。

書き換え環境:書き換え要因と考えられる情報や規則適用制御に関するヒューリスティックスなど多様な情報が、属性-値の形式で設定する。

規則適用制約:書き換え規則の適用条件の一つであり、各書き換え規則が入力をもつどのようなタイプで、どのような方向に書き換えるかという規則の性質や、規則がどのような状況(環境)で適用されるべきかという制約を記述する。

書き換え環境で、書き換え規則の適用制約が満た

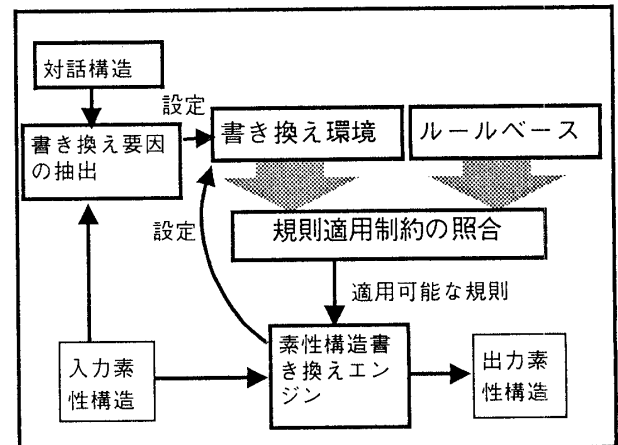


図1. 書き換え環境と適用制約による規則の制御

された時、その規則は適用可能となる。したがって、書き換え環境の状態が変化することにより、その時点における書き換え環境の状態に応じて、一定の性質を持った書き換え規則の集合に、適用可能な書き換え規則が限定され、入力素性構造は特定のタイプにおいて特定方向に書き換えられる。

また、特定の構造を持った意味構造が入力されたとき、その部分構造に対して書き換えタイプと方向を限定して、変換すべき要求が生じることがある。このような場合、規則内で局所的に書き換え環境の状態を設定し、書き換え呼び出しを行うことによって、目的とする構造を得ることができる。このように、ある規則から直接に規則(集合)を呼び出すことはなく、書き換え環境を変化させることによって規則の適用を制御し、規則間の相互依存性を低く保つことが可能になる。

4. 変換過程における書き換え規則の例

つぎに、書き換えシステムを用いてどのように日本語依存意味構造が変換されるかを述べる。

書き換え規則1は、入力を日本語内でイベント的な表現にパラフレーズを行う規則である。

書き換え規則1:

```

on <reln> する-1 in :Lang :japanese
  in= [[reln する-1]
        [obje ?obje]
        ?rest]
  -> ?obje with :e-or-o :event :Lang :Japanese
  ?obje == [?rest]
  out= ?obje
end
  
```

書き換え規則の適用条件は、素性のパスの制約、適用制約、対象の構造的制約の三種類に分類され、

これらが満たされたとき書き換えが行われる。書き換え規則は、素性のパスに対して定義され、“<reln>”のように素性のパスの制約として記述される。書き換え規則1は概念間の意味的關係を表現する素性(RELation)を持つ素性構造に対して定義され、その素性のパスの値が「する-1」である素性構造に対して適用が試みられる。“in :Lang :Japanese”は規則適用制約であり、書き換え環境が日本語内でのパラフレーズを行うような状態に設定されているとき、適用可能になる。“in =”に続く記述は対象の構造的制約であり、入力素性構造との照合により検査される。

(例) 会議への登録をする

文(例)は、素性構造1のように表現される。

(素性構造1)

```
[[reln する-1]
 [agen [[label *spaker*]]]
 [obje [[parm !x <登録-1>]
        [restr [[reln の-adjunct]
                  [arg1 !x]
                  [sloc <会議-1>]]]]]]]
```

書き換え環境が日本語内でのパラフレーズを行うような状態に設定されているとき、素性構造1に関して、書き換え規則1の素性のパスの制約、規則適用制約、構造的制約の三つの制約が満たされ、書き換え規則内の書き換え手順が実行される。

書き換え規則1中の“-> ?obje with...”は、入力の対象格である部分素性構造に対して、書き換え呼び出しによって、書き換えを行っている。“with”に続く記述は、局所的環境設定であり、書き換え呼び出しの際に局所的に書き換え環境が変更される。部分構造の書き換えにおいて、適用可能な規則は、イベント的に書き換えを行う規則の集合に限定され、可能ならば部分構造をイベント的な表現に日本語内でパラフレーズした構造が得られる。

書き換え規則2:

```
on <restr reln> の-Adjunct
  in :Lang :japanese :E-or-V :Event
  in= [[parm ?X]
        [restr [[reln の-Adjunct]
                  [arg1 ?X]
                  [sloc ?Y]]]]
  -> ?X
  ?X.sloc = ?Y
  out= ?X
end
```

書き換え規則3

```
on <restr reln> 登録-1
  in :Lang :japanese :E-or-V :Event
  in= [[parm !X]]
        [restr [[reln 登録-1]
                  [entity !x]]]]
  out= [[reln 登録する-1]
         [agen []]
         [obje []]
         [sloc []]]
end
```

書き換え規則2、3は、日本語内でオブジェクト的表現からイベント的表現に書き換える規則であり、「会議への登録」に対応する意味構造の各要素の書き換えを行う。書き換え規則1では、書き換え規則2、3の適用の結果得られた構造と任意格との構造

を調節し、“out= ...”によって最終的に入力素性構造の書き換えが行われ、素性構造2が得られる。

(素性構造2)

```
[[reln 登録する-1]
 [agen [[label *speaker*]]]
 [obje []]
 [sloc <会議-1>]]]
```

5. 変換過程におけるサブプロセスの生成

書き換え環境の状態を意図的に変化させることにより、書き換えのサブプロセスを生成し、段階的に入力を書き換えて行くことができる。変換過程では、一つのヒューリスティックスとして、日本語内でのパラフレーズ、日英の変換、英語内でのパラフレーズの三つのサブプロセスに変換過程を分割し、段階的に処理することができる。

また、意味構造を構成する概念間の日英の対応関係に基づいて四つのレベルに分類でき、各レベルに応じて書き換え規則を定義している。

(1)universal:概念が言語に依存しない共通のものである。概念の変換処理を必要としないので、これに分類される書き換え規則は存在しない。発話の力のタイプなどがこれに含まれる。

(2)equivalent:概念は言語に共通であるが、特定の領域において成り立つものであり、専門用語などの書き換え規則がこれに含まれる。

(3)similar:概念は特定の言語に依存しているが、状況などを特定することにより、ほぼ共通性がみられるものである。多くの一般用語に関する書き換え規則がこれに含まれる。

(4)exceptional:個々の概念間の関係が直接対応しないが、特定の目的を達成するためには、対応が可能な関係である。表現の間接性などの違いによる概念の対応づけを行う規則などが含まれる。イディオムに関する規則もこれに分類される。

日英の変換を行うサブプロセスでは、exceptional, similar, equivalentの順に優先的に適用を行うため、更に三つのサブプロセスに分割している。これは、exceptional, similar という適用条件の厳しい規則から優先的に適用することにより、規則の競合を少なくし、規則適用の組合せによる書き換え結果数の増加を抑制している。

6. おわりに

素性構造の書き換えシステムにおける書き換え規則の適用制御の枠組みについて述べた。また、書き換えシステムを音声・言語日英翻訳システムの変換過程へ応用した方法について述べた。

謝辞

有益な助言を頂いた言語処理研究室、データ処理研究室の諸氏に感謝いたします。

参考文献

- [1] 長谷川,小暮, 1988, “日英対話文翻訳における意味構造変換手法”, 情報処理第38回全国大会
- [2] 飯田、長谷川、上田、相沢, 1989, “SL-TRANS における変換、生成手法”, 情報処理第39回全国大会
- [3] Nakamura, J., Tujii, Nagao, 1984, “Grammar Writing System (GRADE) of Mu-Machine Translation Project and its Characteristics”, *Proc of COLIG84*