

日英主体表現パターン対を用いた主体表現の日英翻訳とその有効性

山田 武史

宮崎正弘

新潟大学大学院自然科学研究科

1 はじめに

日英翻訳システムにおいて、よりきめ細やかな日英翻訳を行うには、日本語文を、話者の感情・意思などの主観的表現を直接表現した主体表現と、対象を概念として捉えて表現した客体表現に分離し、客体表現を単文変換部を通して英語文として出力する必要がある。本稿では、このような考えに基づき、日本語単文から分離された主体表現をその属性ごとに主体表現属性パターン化し、これを変換された英文に融合することによって様々な英語の主体表現を生成する方法を提案し、実験を通じて、その有効性を示す。

2 主体表現の分類体系

言語表現は、以下のような主体表現と客体表現で構成されている [1]。

主体表現：話者の感情・意思などの主観的表現を直接表現したもの。日本語では、助動詞や助詞など。

客体表現：対象を概念として捉えて表現したもの。日本語では名詞、動詞、形容詞、副詞など。

さらに、主体表現を以下の5つに大分類した。

判断：客体認識に対する話者の直接的な判断を表す。肯定・否定など。

態：主体となんらかの作用との関係を表す。使役・受身など

時制：話者が表現の焦点をどの時間においているかを表す。現在・過去・未来など。

相：動作や状況のあり方を表す。進行・完了など。

様相：客体認識世界に対する話者の主観を表す。推量・要求など。

なお、時枝・三浦文法では、態と相は客体表現に分類されているが、これらは話者に共通するごく少数の属

性のセットで表されることから、主体表現に含めて扱うこととした。

さらに各大分類を中分類・小分類・細分類からなる最大3階層に細かく分類し、ノード数約150の4階層からなる図1に示すような主体表現属性体系を構築した。

様相属性の階層の分類に関しては、同一単文内に同時に存在しない様相属性の排他性を考慮したグルーピングを行った。

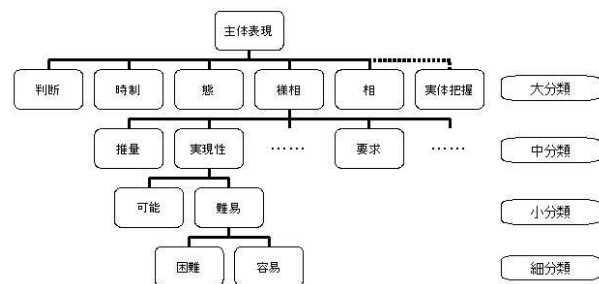


図 1: 主体表現属性体系（一部）

3 主体表現/客体表現分離融合方式

言語表現は主体表現と客体表現が混合された形態をとり、日本語と英語においても、言語族の違いによる認識の違いが反映されているため、表現の対応関係をとるのが困難である。そこで、言語の違いによる話者の見方の違いを取り出し、対象の共通性と言語による見方の違いを分けて処理するためには、まず、主体表現と客体表現を分離し、別々に処理することが有効である。[2]

また日本語は、主体表現に助詞、助動詞などの単語が用いられるのに対して、英語は助動詞や前置詞などの

単語だけでなく語の屈折（言語変化）を伴って主体の直接的表出が行われることが多い。従って、日本語の主体表現の語と英語のそれとは直接的に対応しないことが多い。

一方、客体表現は、それぞれの言語の様々な固有な表現を持ち、その内容も実に多様である。一方、主体的表現は表層的表現は多様であるが、その内容は話者の主観的判断や感情に関する情報であり、極めて限定されている。従って、主体的表現はパターン化して、素性として扱うことが可能であり有効である。

そこで、日本文に表わされた話者の認識を解析するに当たって、話者の主観的感情や意志を分類し、与えられた日本文の主体的表現の部分が、どのような感情や意志を表わしているかを解析する。

この過程で、もとの日本文は客体表現情報と主体表現情報に分離される。主体表現情報は目的言語の主体表現に訳出できる範囲の分解能と精度で分類すれば良く、このため、その他の情報は客体表現の構造を示す情報として保存し、パターン変換の枠組みを経由して目的言語に変換する。変換対象となる客体表現には、文要素の格を表示する格助詞および格助詞相当語が含まれることになる。この客体表現は変換後、英語の客体表現（英語素文）となる。そのあと、既に抽出されている話者の感情や意志が英語素文に対して組み込まれる。この組み込みでは、助動詞や前置詞の挿入の他、種々の語の屈折（変形）が行われる。このようにして、日本文中から分離された主体表現情報が英文生成の段階で英文に融合される。

図2に主体表現/客体表現分離融合方式による翻訳処理例を示す。

4 英文生成における主体表現の生成

4.1 主体・客体分離・翻訳プログラム

英文生成プログラムのための、上位プログラムである、主体・客体分離・翻訳プログラムの概要に触れる。このプログラムは、入力された日本語文を、日本語形態素解析システム Maja[3]を用いて素解析し、それに後処理を施した後、主体表現と客体表現をそれぞれ取り出し、客体表現を用例翻訳する。主体表現を属性ごとに配列に入れ、客体翻訳を NP と VP に分け配列に入れ、それぞれ、配列 sub_info(主体

表現情報)、obj_english(客体英文)として出力する。英文生成プログラムは、この主体・客体分離・翻訳プログラムの出力を入力とする。

4.2 日英主体表現パターン対

主体表現生成用日英主体表現パターン対の構成を以下に示す。

[”中分類属性”, ”小分類属性”, ”細分類属性”, ”英文パターン”, ”動詞情報”, ”パターン内動詞”]

・中、小、細分類:

主体表現の分類体系の中、小、細分類にあたる属性
・英文パターン: 主体表現の分類体系の葉にあたる属性に対応する英文パターン

・動詞情報: 英文パターンによって VP 部に及ぼす動詞の語尾変化情報

・パターン内動詞: 英文パターン内の動詞

助動詞パターン15、助動詞以外のパターン47の計62のパターン対が作成されている。作成したパターン対の例を以下に示す。

```
$modality_db =
[["実現性", "可能", "-", "NP be able VP",
  "VP.inf", "be"],
["評価", "-", "-", "NP be worth VP",
  "VP.gnd", "be"], ...]

$auxiliary_verb_db =
[["推量", "確実な推定", "-", "NP must VP", "-"],
["実現性", "可能", "-", "NP can VP", "-"], ...]
```

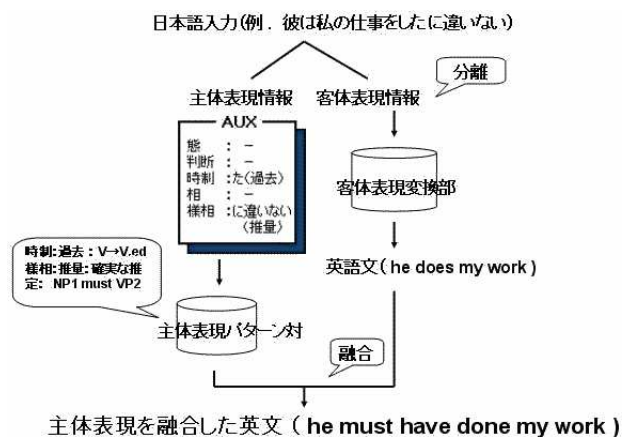


図2: 主体表現/客体表現分離融合方式による翻訳例

4.3 主体表現生成処理の流れ

処理は以下の方針をとり、1～6の順に行う。

方針：単文の型「OM1 SYN1 NP AV TE1 JU1 ADV TE2 MO JU2 VP AS OM2 SYN2 QUE」の各要素に合う適当な英文パターンを、入力された主体表現情報から生成し、それを型の各要素に代入し、英文生成を行う。型の各要素の定義を以下に示す。

NP：noun phrase（名詞句）

VP：verb phrase（動詞句）

TE：tence（時制）

JU：judge（判断）

AV：auxiliary verb（助動詞）

ADV：adverb（副詞）

MO：modality phase（様相句）

AS：aspect（相）

SYN：syntax（構文）

OM：other modality（その他の様相表現）

QUE：question（疑問符）

1. 主体情報、客体英文を入力し、ハッシュ生成
2. 主体表現パターン対と主体表現情報を照合、抽出
3. 形態素調整等の文法処理を行う
4. ハッシュに、AV、MOなど、属性名がキーの要素の値に、生成した主体表現パターンをストアする。
5. 動詞情報に従い、形態素調整を行う。
2～5を、各属性において行う。
6. 単文の型の各要素に、生成したハッシュの同名キーの値を代入し英文を生成する。

4.4 主体表現生成処理の例

全ての属性を網羅した処理を行いたかったので、不自然な破綻した日本語であることを断っておく。

```
mirage{take}54% emake.rb
```

入力日本語文：

コップから牛乳が溢れられる価値がある必要があることがあり始めるかもしれないこともなかった

客体翻訳（入力1）

```
[["The glass", "NP"],
```

```
["overflow with milk", "VP"]]
```

主体表現情報（入力2）

```
["主体判断", ["判断", ["否定", "二重否
```

```
定"]], ["時制", "過去"], ["態", "受動・使役／受動"], ["様相", ["推量", "不確実な推定"], ["実現結果", "頻度", "頻度有"], ["評価", "価値あり"], ["義務性", "必要"]], ["相", "開始"]]
```

入力客体表現からハッシュ型のデータ配列を作る

ハッシュデータは

```
{"TE1"=>"", "QUE"=>"", "TE2"=>"", "MO"=>"",  
"NP"=>"The glass", "VP"=>"overflow with milk",  
"JU1"=>"", "AS"=>"", "JU2"=>"", "ADV"=>"",  
"OM1"=>"", "OM2"=>"", "SYN1"=>"", "SYN2"=>"",  
"AV"=>""}
```

英文の型は、

```
e_form = "OM1 SYN1 NP AV TE1 JU1 ADV TE2 MO  
          JU2 VP AS OM2 SYN2 QUE"
```

1. 様相表現処理部

1-1. 構文処理

→構文表現なし

1-2. 助動詞処理

ハッシュを更新

```
{"AV"=>"might"}
```

1-3. 助動詞以外の様相表現を捕まえる

```
{"MO"=>"be worth needing ",  
"VP"=>"to overflow with milk ",  
"ADV"=>"occasionally"}
```

2. 判断処理部

```
{"MO"=>"being worth needing ",  
"JU1"=>"not without "}
```

3. 態処理部

```
{"VP"=>"to be overflowed with milk"}
```

4. 相処理部

```
{"TE1"=>"", "QUE"=>"", "TE2"=>"",  
"MO"=>" start being worth needing"}
```

5. 時制処理部

```
{"TE2"=>"have",  
"MO"=>" started being worth needing"}
```

処理後の英文は

```
"The glass might not without occasionally  
having started being worth needing to be  
overflowed with milk"
```

5 評価

プログラムを通して出力された英文が実際使われているような英語表現かを確かめる実在性テストを google によるインターネットコーパスを用いて行い、検証した。

日経コーパスから単文を取り出し、動詞が、自動詞の場合は「溢れる」、他動詞の場合は「掛ける」に変えた（上位システムの試作中の日英翻訳システムが、用例翻訳によって翻訳行っているため）試験文を200文用意する。また、独自に、[4]を参考に、主体表現の組合せを網羅するような例文を独自に100文を作成し、これら300文を試験文とした。

まず、これら例文を試作中の英文翻訳プログラム、および英文生成プログラムに通し、英文出力を得る。この英文を、googleで検索にかけ、ヒットすれば、その英文が世界で使われていることがわかる。このことによって、実際使われているような英文が生成されたと判断する。

結果として、日経コーパスの例文200文中、146文がインターネットコーパスにマッチし（実在率73%）、実在性の高い英文が生成されたことが証明された。

また、論文を基に、主体表現をできるだけ網羅するよう作成した日本語文は、100文中60文がインターネットコーパスにマッチし、こちらの結果からも、比較的実在性の高い（実在率60%）英文が生成されたことが証明された。

使用試験文例。

コップから牛乳が溢れる。

彼はやかんを火に掛ける。

コップから牛乳が溢れるだろう。

彼はやかんを火に掛け終える。

コップから牛乳が溢れ続けている。

彼はやかんを火に掛けたい。

彼はやかんを火に掛けてみる。

コップから牛乳が溢れない。

彼はやかんを火に掛けない。

彼はやかんを火に掛けないだろう。

コップから牛乳が溢れないに違いない。

彼はやかんを火に掛けないに違いない。

コップから牛乳が溢れた。

彼はやかんを火に掛けた。

コップから牛乳が溢れたかもしれない。

彼はやかんを火に掛けたところだ。

コップから牛乳が溢れ始める。

etc...

6 おわりに

本稿では、分離された主体表現を日英主体表現パターン化し、これを変換された英文に融合することによって様々な主体表現を生成することを可能とし、その有効性を示した。

また、一連の処理は、ルールベースで行っているため、データの追加・修正が可能である。

今後の課題には以下のようなものがある。

- ・動詞の語尾変化情報などの形態素調整
- ・更にきめ細かい、実用英語に近い翻訳の実現
- ・複文の翻訳

現在、これらを解決し、発展させていくために、議論及び、実装を進めている。

なお、本稿で対象外とした日本語の係助詞・副助詞など客体表現の構成要素となり、実態の捉え方を表す。主体表現（図1の主体表現属性体系において大分類「実体把握」として拡張予定）については、今後検討する必要がある。

参考文献

1. 三浦つとむ：日本語とはどういう言語か、講談社学術文庫、(1976)
2. 池原悟，宮崎正弘，白井諭，林良彦：言語における話者の認識と多段翻訳方式，情報処理学会論文誌，Vol.28,No.12, pp.1269-1279 (1987)
3. 日本語形態素解析システム Maja:
<http://www.nlp.ie.niigata-u.ac.jp/nlp/maja/>
4. 池原悟，宮崎正弘，白井諭：言語過程説から見た日本語述語の構造，自然言語処理の新しい応用シンポジウム論文集，PP.141-142(1992).