コミュニティ型機械翻訳サイト「訳してねっと」の基盤技術とその展開

北村美穂子　村田稔樹　介弘達哉　下畑さより　佐々木美樹　松永聡彦　中川哲治
沖電気工業株式会社　研究開発本部

1. はじめに

インターネットの発達に伴って機械翻訳システムの需要は高まり、現在では数多くの翻訳サービス、翻訳ソフトが存在する。しかし、インターネット上の様々な分野のページに対して、機械翻訳システムが精度良く翻訳するためには、翻訳する分野に応じた辞書の増強、および、恒常的な辞書の管理が不可欠になる。

我々は、この課題を解決するために、ユーザがインターネットを通して分野毎に協力して辞書や文書を登録することによって翻訳品質を高めるWebベースのコミュニティ型機械翻訳翻訳サイト「訳してねっと[1]」を提案した。現在、本サイトを限定的にユーザに公開し、英日・日英翻訳の運用及び実験を行っている。

本稿では、まず、その基盤技術であるパターンベース翻訳方式について説明し、次に、本方式の特長を活かしたコミュニティ型機械翻訳サイト「訳してねっと」の基本思想及びその機能について紹介する。最後に、パターンの自動獲得技術や多言語化などの今後の展開について述べる。

2. パターンベース翻訳方式

パターンベースの翻訳システム(1)は、解析、変換、生成処理を翻訳パターンだけで行う。翻訳知識はすべて翻訳パターンで記述されているため、翻訳知識の可読性に優れており、ユーザによる翻訳パターンの追加も容易である。しかし、既存のパターンベース翻訳方式では、非終端記号に意味などの条件を与えるためには意味毎に非終端記号を用意する必要があったり、素性の扱いが限定されるなど、ユーザが自由にパターンを作成するには課題が多く、システム内を熟知していないと翻訳パターンの追加登録は難しいという問題があった。一方、素性の単一化が可能なHPSGパーザは、計算量が多いという問題があり実用化が難しい。

我々は、意味や人称など非終端記号や単語が持つ種々の条件はすべて素性(素性名と素性値の組合せ)で与え、素性の単一化ではなく素性の一致や継承を可能にすることで、ユーザの可読性に優れた実用レベルのパターンベースの機械翻訳システムを実現した。これによりユーザはシステム内の翻訳処理を知らなくても、副作用を及ぼすことなく翻訳パターンを追加することができる。

さらに、素性の継承は単一化ではなくコピーで実現し、非終端記号が有する素性を制限することで候補を大幅に削減する機構や、解析失敗の救済機構や優先度制御の機構も兼ね備える。以下にそれらについて説明する。

・失敗救済機構を備えた多段階辞書適用方式

図１に本システムの構成図を示す。まず入力文が形態素辞書を用いて形態素解析されると、結果は構文解析・生成モジュールに渡される。構文解析・生成モジュールはユーザが設定した複数のユーザ辞書を優先度の高いものから順に適用して、形態素解析結果をボトムアップに解析する。ユーザ辞書中に適用可能な翻訳パターンが存在しない場合は、汎用的な語彙で構成されるシステムが持つ辞書(システム辞書)を適用する。またシステム辞書中にも存在しない場合は失敗救済辞書を適用する。失敗救済辞書とは、通常の文法規則では解析不可能な文でも、尤もらしい翻訳結果を出力するために用意される辞書であり、例えば、主格と主辞の動詞の人称数情報の不一致を許す翻訳パターンなどが登録されている。こ

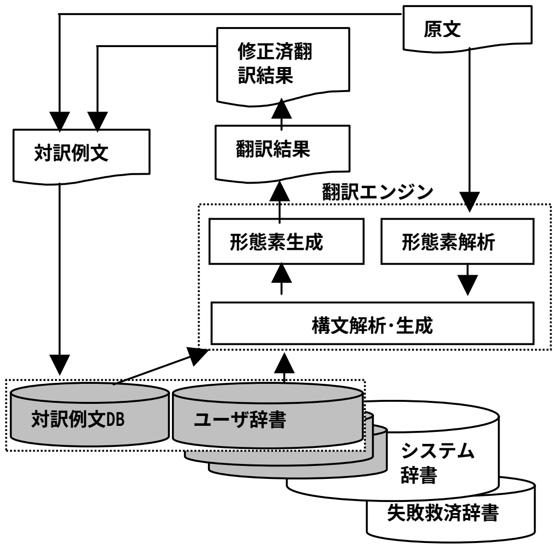

図１ パターンベース機械翻訳システムの概観

Technology and Development on Collaborative Translation Environment "Yakushite Net"
Mihoko KITAMURA, Toshiki MURATA, Tatsuya SUKEHIRO, Sayori SHIMOHATA, Miki SASAKI, Toshihiko MATSUNAGA and Tetsuji NAKAGAWA ,Oki Electric Industry Co., Ltd., Corporate Research & Development Center

[1] http://www.yakushite.net/

のように辞書の重要度に応じて、多段階に辞書を適用することで、ユーザが指定する辞書は最優先に適用され、かつ、文法的に正しくない文の入力においても解析失敗とならずに翻訳結果が得られるというロバストな解析を実現している。さらに、適用される翻訳パターン候補を最小限に抑えながら解析するため処理も高速になる。

パターンベース翻訳方式では、解析が成功すると同時に目的言語の生成結果を得ることができる(次項を参照)。生成結果は形態素生成辞書を用いて語形変化、活用形などが調整され、ユーザは出力結果を得る。出力結果は、ユーザにより修正が加えられ、対訳例文として、データベースに格納される。この対訳例文は、以降、同一文が入力された場合に翻訳メモリとして利用される。

・柔軟な素性操作をもつ翻訳パターン

英日翻訳で用いられる翻訳パターンの例を図２に示す[1]。図２の(1)から(7)が語彙に関するパターンであり(一般的には辞書と呼ばれる)、(8)から(12)が文法に関するパターンである。図２に示すように翻訳パターンにおいては文法、辞書といった特別な区別はなく、すべて統一的な形式で記述する。

図２の(1)を一般的な書き換え規則で表すと

play NP→VP:VP←NP を 弾く

となり‵en′で始まる英語パターンと‵ja′で始まる日本語パターンがシンクロナイズしている。ここでは「解析が成功する」とは、形態素解析結果を原言語の‵en′で始まる翻訳パターンにボトムアップに適用させて、最終的に非終端記号“S”(図３の(12))の適用に成功することを言う。非終端記号“S”の適用に成功すれば、原言語の解析結果に対応する目的言語側‵ja′の解析結果をトップダウンに生成し、生成結果、つまり翻訳結果を得ることができる。

図２にみるように翻訳パターンは終端記号、非終端記号共 pos=v(品詞=動詞), personNum=3sg(人称・単複=３人称単数)など、複数の素性を与えることができる。sem=human|org(意味=人間または組織)のように１つの素性に対し複数の素性値を与えることもでき、pos!=v のように記述することによって否定の情報を与えることもできる。

原言語パターンの右辺[2]の素性は制約を表し、入力文の形態素解析結果が有する素性がパターンの素性の制約を満たせばパターン適用は成功する。左辺の素性は付与を表し、パターン適用が成功した時点で、左辺の非終端記号にその素性が与えられる。この原言語側の素性の制約を利用することによって、

```
(1)  [en:VP:sSem=human play:pos=v:*
           [1:NP:sem=instrument]]
     [ja:VP [1:NP] を:pos=particle 弾く:pos=v:* ];

(2)  [en:VP:sSem=human play:pos=v:*
           [1:NP:sem=sport|game]]
     [ja:VP [1:NP] を:pos=particle  する:pos=v:* ];

(3)  [en:VP:sSem=music|instrument play:pos=v:* ]
     [ja:VP 鳴る:pos=v:* ];

(4) +[en:VP:sSem=human play:pos=v:* ]
      [ja:VP 遊ぶ:pos=v:* ];

(5)  [en:N piano:pos=n:sem=instrument:*]
     [ja:N ピアノ:pos=n:* ];

(6)  [en:N tennis:pos=n:sem=sport:*]
     [ja:N テニス:pos=n:* ];

(7)  [en:N Ken:pos=n:sem=human:*:personNum=3sg]
     [ja:N 健:pos=n:* ];

(8)  [en:SentenceSub since:pos=conj [1:Sentence:*]]
     [ja:SentenceSub [1:Sentence:sentenceType=sub:*]
           で:pos=particle];

(9)  [en:Sentence [1:NP:sem={SEM}:personNum={NUM}]
           [2:VP:sSem={SEM}:personNum={NUM}:*]]
     [ja:Sentence:setenceType=main [1:NP]
           は:pos=particle [2:VP:*]];

(10) -*[en:Sentence [1:NP:personNum={NUM}]
           [2:VP:personNum={NUM}:*]]
        [ja:Sentence:setenceType=main [1:NP]
           は:pos=particle [2:VP:*]];

(11) [en:Sentence [1:NP:sem={SEM}]
           [2:VP:sSem={SEM}:*]]
      [ja:Sentence:sentenceType=sub [1:NP]
           が:pos=particle [2:VP:*]];

(12) [en:S [1:Sentence:*] ]
      [ja:S [1:Sentence:sentenceType=main:*]];
```

図２ 翻訳パターンの例

図２の(1)(2)の例 “play”を「ピアノを*弾く*」「テニスを*する*」のように目的格に位置する名詞の意味により訳し分けることができるようになる。

一方、目的言語パターンはその逆で、左辺の素性は生成時の制約となり、右辺の素性は付与を表す(図２の(10)(11)の例)。この目的言語側の素性の制約を利用することによって、「私*が*ピアノを弾く時」、「私*は*ピアノを弾く」といった構文上の違いにより訳し分けることができるようになる。

さらに、非終端記号や終端記号が持つ素性間の参照も可能にする。(9)の “[1:NP:sSem={SEM}]”と “[2:VP:sSem={SEM}]” の “{SEM}”は、「主格に位置する名詞の意味属性と動詞が有する主格の意味属性は共通する」ことを表している。これにより主格が「ピアノ」の場合は、図２の(3)が適用され「ピアノが*鳴る*」となり、主格が「健」の場合は図３の(4)が適用され「健が*遊ぶ*」となる。この記法により素性の参照による訳し分けを可能にし、英語文法における人称や単数・複数情報の一致の問題も簡単

[1] 説明のために実際のパターンより簡略化している。

[2] 図３の(1)の場合、en:に続く VP が書き換え規則の左辺であり、play や 1:NP が右辺となる。

な記述によって解決することができる。

　また、素性の継承方法にも特長を有する。図２の‶play:pos=v:*″等にみられる‵*′の記号は主辞を表し、右辺の ‵*′を持つ終端記号または非終端記号の素性が左辺の非終端記号に継承される。ただし、図３のように記述された素性定義テーブルを利用することによって、各非終端記号がとり得る素性を限定している。

```
Sentence = {  sentenceType  };
VP       = {  personNum
              conjugation
               sSem         };
NP       = {  personNum
               sem          };
```

図３ 素性定義テーブルの例

　これは、非終端記号 ‶VP″(動詞句)の素性の一つである conjugation(活用形)は非終端記号‶Sentence″(文)の書き換え規則では不必要であるため、継承させないというものである。この機構により各非終端記号は必要な素性のみを持つことになり、翻訳パターンの可読性を高め、同じ素性をもつ非終端記号をまとめる(選言的な解析結果をまとめる)ことができ解析結果候補を削減することができる。

・同一辞書内での翻訳パターン適用優先度制御

　翻訳パターン方式では、条件の厳しいパターンのみの翻訳では翻訳失敗が起こりやすく、逆に条件が緩いパターンを利用すると過適用になり候補が爆発するという問題がある。これを回避するために、次に述べるパターンの優先度制御方式を考案した。

　図３の(9)は動詞の訳し分けのために素性の参照によって主格の意味を制限した翻訳パターンである。しかし、もしユーザが正しく意味を付与していなかった場合には制約を満たせず翻訳が失敗する。ユーザの付与ミスを救うためには、意味を制限しない(10)のような翻訳パターンも必要となるが、このパターンを登録すると、(9)(10)の両方のパターンが適用されてしまい、候補が倍増する。さらに、パターンに優劣がないと、解析結果の優劣が判断できず、尤もらしい翻訳結果を１つだけ得ることができない。上記の問題を回避するために、(10)のようにパターンの前部に ‵*′記号を付与することによって、詳細な条件を持つパターンがマッチしない場合のみ適用される機能を設け、パターンの過適用を避ける。さらに、パターンに ‵+′プラスと ‵-′マイナスを付与することにより、プラスを持ったパターンは、コストが低くなり、マイナスを持ったパターンはコストが高くなるように設定する。例えば ‶Ben plays.″で Ben が未知語でその意味が不明の場合は、(3)と(4)の両方が(10)に適用されるが、その場合は‵+′を持つ「遊ぶ」(汎用的な訳語)が優先され、「Ben は遊ぶ。」となる。さらに、入力文の解析時に適用した翻訳パターンの数もコストに反映させ、総合的な解析結果のコスト計算を行っている。

　上述した我々のシステムの特長をまとめると、ユーザによる辞書管理が容易である、ユーザが登録する辞書を最優先に適用する機構を持つ、ユーザは単語やイディオムだけではなく「次に～を示します⇔following is ～」のような言語固有の言い回しや表現レベルの辞書も登録できる、という点である。つまり、本システムは、辞書の作成、管理をユーザに開放することができるユーザ主導開発型の機械翻訳システムということができる。

３．コミュニティ型機械翻訳サイト 「訳してねっと」

　インターネット上では不特定多数の人々が「コミュニティ」と呼ばれるグループを形成し、議論や情報交換、さらには共同作業をしている。翻訳に関して言えば、翻訳のプロではない各技術の専門家が海外の技術的な記事やマニュアルを翻訳するなど、ボランティアベースの翻訳サイトが数多く存在する。彼らはインターネットを通じて多数の仲間を募り、翻訳作業を協力して行っている。

　我々は、このようなインターネット上でのユーザの協調作業環境と、先に述べたユーザ主導開発型の機械翻訳システムを融合させることにより、インターネットユーザが仲間同士で辞書を作成、共有したり、さらには機械翻訳結果を仲間同士で修正しあったり、管理することができる機械翻訳システムの開発を目指した。本システムをコミュニティ型機械翻訳システムと呼ぶ[(2)]。

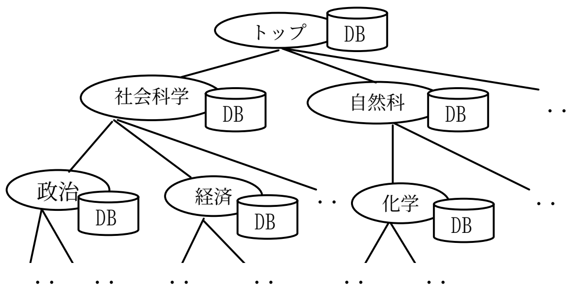

図４ 階層ツリー構造に分類されたコミュニティ

　コミュニティ型機械翻訳システムは図４のように階層ツリー構造に分類された分野（これをコミュニティと呼ぶ）を多数有している。これは様々な分野の仲間同士が個々に作業するための環境を提供する。図４の例では社会科学コミュテニィの下位に政治、経済というコミュニティが存在し、政治や経済の下位にもコミュニティが存在する。このようなコミュニティはユーザが自由に作成することができる。各コミュニティはコミュニティのメンバーが作成、管理するコミュテニィ辞書や過去の翻訳結果を有した

対訳データベースを持つ。コミュニティ内で翻訳する際には、自らのコミュニティ辞書を最優先に利用し、直近の親からトップまでのコミュニティ辞書を優先度を順に下げて利用する。なお、これらのコミュニティ辞書は、図１のパターンベース機械翻訳システムにおいては多段階で適用されるユーザ辞書に相当する。また、トップの辞書は、どの分野でも利用できる汎用的な辞書、つまり、図１のシステム辞書に相当する。辞書や翻訳結果の作成、管理はコミュニティ単位で行われ、その利用はその下位に存在する分野関連度の高いコミュニティに限られている。これによりユーザは関連分野でないコミュニティが及ぼす翻訳結果への影響を気にすることなく、その分野で利用される専門用語や固有表現を登録することができる。

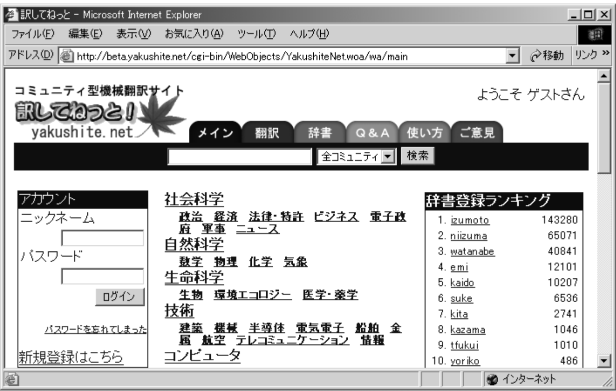

図５ 訳してねっとのトップ画面

これらの機能を搭載した「訳してねっと」のトップ画面を図５に示す。図中央に社会科学、政治などのコミュニティが表示されているが、これらをクリックすると各コミュニティに入ることができる。各コミュニティに入ると、コミュニティ名が画面上部に表示され、以降は翻訳タブや辞書タブをクリックするだけで各々の機能をそのコミュニティ環境で利用することができる。

各コミュニティには「Ｑ＆Ａ」(質問応答集)、「関連ページリンク集」(各コミュニティに関連する Web ページへのリンク作成機能)など、コミュニティ活動の活性化を促す各種機能を持つ。さらに、図５の画面右側にみるように、辞書登録ランキングを表示することによって辞書登録を促す工夫をしている。コミュニティメンバーはこれらの機能を利用して協調翻訳を容易に行うことができる。さらに一般のユーザでも蓄積されたコミュニティ辞書を利用することによって分野に合った高品質な翻訳結果を得ることができる。

4．今後の展開

パターンベース方式を採用した理由の１つとして、統計ベースの翻訳知識獲得技術との親和性が高いことが挙げられる。「訳してねっと」では、ユーザは既存の対訳例文や機械翻訳結果を、図１の対訳例文データベースに登録するが、その対訳例文データベースから翻訳知識獲得技術[3][4]を利用することにより、対訳辞書や翻訳パターンを自動的または半自動的に作成することができる。作成、利用される翻訳パターンは人間にとって理解しやすい形式であるため、翻訳パターンの修正が可能である。そうして作成した辞書を各コミュニティ辞書として翻訳処理に利用することにより、分野や文書に依存した表現を多く含んだ文書の翻訳品質を高めることができると考える。さらに、専門用語の自動抽出機能[5]を備えることにより、翻訳品質向上に必要とされるユーザの負担を軽減することができる。

上記の翻訳環境の実現は、近年その技術開発が望まれている多言語翻訳環境にも関係する。通常、新しい言語対の機械翻訳システムの構築には、対訳辞書や原言語と目的言語の文法規則からなる翻訳規則を作成しなければならない。しかし、パターンベース翻訳方式では、新しい言語対の対訳例文があれば、自動獲得技術によって翻訳パターンを自動作成し、その翻訳パターンを用いて機械翻訳することができる。利用初期の段階では部分的な翻訳しかできないと思われるが、各言語に精通した人間の翻訳を支援する形態で利用していき、対訳例文が増え、翻訳知識が蓄積されれば、将来的には自動翻訳も可能になると考える。しかし、多言語化のためには、多言語データベースをどのように構築するか、形態素解析．生成処理をどうするかなど、解決すべき課題も多い。

今後も、上記のような多言語化の課題に関する研究開発、及び、学習機能とユーザとの協調作業によって翻訳知識を増強する機械翻訳環境の研究開発を一層進めていきたい。

なお、本研究は通信・放送機構平成 14 年度基盤技術研究促進制度に係る研究開発課題「多言語標準文書処理システムの研究開発」の一環として行われている。

(1) Takeda, K. 1996. “Pattern-Based Context- Free Grammars for Machine Translation”. In proceedings of *the 34th Annual Meeting of the Association for Computational Linguistics*, pp.144-151.
(2) Shimohata, S., Kitamura, M., Sukehiro, T. and Murata, T. 2001. “Collaborative Translation Environment on the Web”. In proceedings of *the MT Summit VIII*, pp.331-334.
(3) Kitamura, M., and Matsumoto, Y. 1996. “Automatic Extraction of Word Sequence Correspondences in Parallel Corpora”. In proceedings of *the 4th Annual Workshop on Very Large Corpora* pp.79-87.
(4) 北村美穂子，松本裕治，1996，“対訳コーパスを利用した翻訳規則の自動獲得”，情報処理学会論文誌，vol.37-6, pp.1030-1040
(5) Shimohata, S., Sugio, T. and Nagata, J. 1997 “Retrieving Collocations by Co-occurrences and Word Order Constraints”. In proceedings of *the 34th Annual Meeting of the Association for Computational* Linguistics, pp.476-481.