文末機能表現シソーラスと述部正規化システム

松木 久幸 (名古屋大学大学院 工学研究科) †
佐藤 理史 (名古屋大学大学院 工学研究科) ‡
駒谷 和範 (名古屋大学大学院 工学研究科) ††

A Thesaurus of Predicate Functional Expressions and its Application to Predicate Standardization

Hisayuki Matsuki (Graduate School of Engineering, Nagoya University)
Satoshi Sato (Graduate School of Engineering, Nagoya University)
Kazunori Komatani (Graduate School of Engineering, Nagoya University)

1 はじめに

日本語には、文末の用言に接続し多様な意味を表す機能表現が多数存在する。これらの表現を**文末機能表現**と呼ぶ。文末機能表現には、ほぼ同じ意味を表す表現が複数存在する。このため、テキストマイニングにおける意見の集約など、文の意味の同一性の判定が必要となるタスクにおいては、同一の意味を持つとみなす文末機能表現を、一つの代表表現に置き換える**正規化** が必要となる [1]。たとえば、あるタスクでは、〈依頼〉の意味[1]を表す「〜テください」「〜テほしい」「〜テくれるか」を、すべて一つの代表表現（たとえば、「〜テください」）に正規化することが求められる。

このような正規化は、それぞれの文末機能表現に代表表現を定義することによって実現することができると考えられる。しかしながら、実際には、問題はそれほど単純ではない。第一に、文末機能表現の単位が問題となる。たとえば、「〜テもらう」と「〜たい」は、それぞれ単独で文末の用言に接続する一方で、結合した「〜テもらいたい」という形でも文末の用言に接続する。単独で用言に接続する場合は、それぞれ〈受益〉、〈希望〉の意味を表すが、結合した場合は〈願望〉の意味となる。このため、「〜テもらう」と「〜たい」に代表表現を定義するだけでは不十分であり、「〜テもらいたい」にも代表表現を定義する必要がある。この問題を解決するために、我々は、用言直後から文末までの長い単位を文末機能表現と捉え、この単位を見出し語とする辞書（シソーラス）を作成する。すなわち、「〜テもらう」と「〜たい」だけを見出し語とするのではなく、「〜テもらいたい」も見出し語に含める。なお、本論文では、「〜テもらう」と「〜たい」のような短い単位を、文末機能表現の構成要素と呼ぶことがある。

第二の問題は、どの範囲の表現を一つの代表表現に正規化すべきかが、応用タスクに依存して定まるという点である。たとえば、事実か推量かの判定だけが必要となるタスクでは、「〜にちがいない」と「〜かもしれない」は、どちらも〈推量〉の意味を表す「〜だろう」に正規化することができよう。一方、推量の確信度も考慮したいタスクでは、これらの表現を一つの代表表現に正規化するのは、明らかに不適切である。この応用タスク依存性の問題を解決するために、我々は、文末機能表現を意味分類した辞書（文末機能表現シソーラス）に代表表現を直接定義するのではなく、ユーザー定義に従って、この辞書から正規化用の辞書を生成するアプローチを採用する。

図１に、作成したシステムの全体像を示す。この図に示す通り、作成したシステムは、３つのサブシステムから構成される。以下、本論文では、これらのサブシステムについて説明する。

†h_matuki@nuee.nagoya-u.ac.jp
‡ssato@nuee.nagoya-u.ac.jp
††komatanii@nuee.nagoya-u.ac.jp
[1]本論文では、機能表現の意味を表す際に、記号 '〈〉' を用いる。

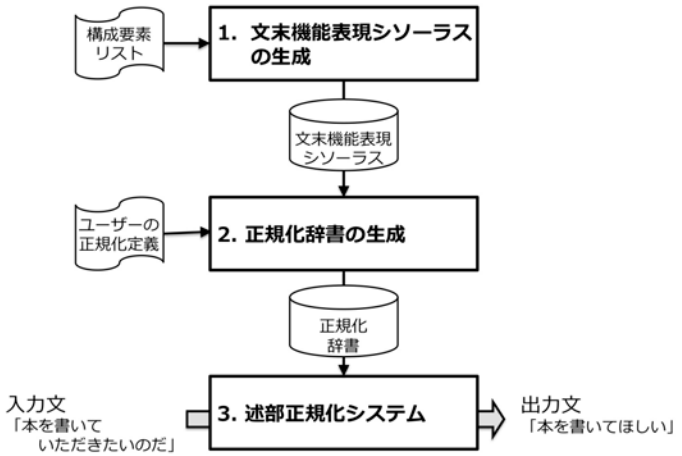

図 1: 本研究の全体像

2　文末機能表現シソーラスの生成

本節では、文末機能表現シソーラスの生成について述べる。このシソーラスの見出し語リストには、以前の研究 [5] に基づいて、作成した文末機能表現のリストを使用する。このリストは、人手で定義した 165 種類の構成要素から合成されている。この見出し語リストに意味表現を付与し、文末機能表現シソーラスを作成する。

2.1　意味体系

文献 [2, 3, 4] を参考に、本シソーラスで使用する意味体系を設計した。この意味体系は木構造の形式をとる（表 1）。この木構造の葉ノードが、意味を表現する最小構成要素であり、これを**意味ラベル**と呼ぶ。意味ラベルは、根ノードからのパス表現で表す。以下に、意味ラベルの例を示す。

(1) 〈/モダリティ/表現類型/意志/つもりだ〉

(2) 〈/モダリティ/表現類型/意志/意志形〉

例 (1) は、機能表現「つもりだ」に付与される意味ラベルで、その意味が、最上位レベルでは〈モダリティ〉に区分され、第２レベルでは〈表現類型〉、第３レベルでは〈意志〉、最下位レベルでは〈つもりだ〉に区分されることを表している。意味体系の最下位レベルのほとんどは、具体的な機能表現（リテラル）に対応する。

例 (2) は、活用形に対応する意味ラベルの例である。このような意味ラベルは、7 種類あり、それらは、タ形、意志形、命令形、推量形、省略意志形、文語命令形、省略推量形に対応する[2]。なお、リテラルと活用形以外の最下位レベルには、〈デス列〉、〈可能形〉、〈否定形〉、〈謙譲語〉の４つがある。

本論文では、以降、意味ラベルの最下位レベルを除いた部分を、略記表現（表 1 の最左列）で表す。たとえば、〈/ヴォイス/可能/ことができる〉を、〈可能/ことができる〉と表す。また、最下位レベルを省略し、〈可能〉と書くこともある。

本シソーラスでは、この意味ラベルを組み合わせて、文末機能表現の意味を表現する。すなわち、文末機能表現の意味表現は、意味ラベル列となる。なお、意味ラベルの接続は、記号 '=' で表す。たとえば、〈開始/テいる〉と〈必然性/はずだ〉の接続は、〈開始/テいる=必然性/はずだ〉と表す。

2.2　意味表現の付与

見出し語に対する意味表現の付与は、以下の手順で行う。

1. 文末機能表現の構成要素に対して、人手で意味表現を定義する。

[2] これ以外の活用形は、意味を持たないものとして扱う。

表 1: 意味体系

上位レベルの略記表現	意味ラベル 上位レベル	最下位レベル
〈 可能 〉	/ヴォイス/可能/	ことができる, こともできる, Rうる, 可能形
〈 受身 〉	/ヴォイス/受身/	(ら) れる
〈 使役 〉	/ヴォイス/使役/	(さ) せる
〈 過去 〉	/テンス/過去/	タ形
〈 否定 〉	/肯否/否定/	ぬ, ない, テない, Rない, 否定形
〈 開始 〉	/アスペクト/開始/	Rはじめる, Rだす, Rかける
〈 継続 〉	/アスペクト/継続/	ている, てる, ていく, てくる, Rつづける, Rつつある
〈 終了-無意志 〉	/アスペクト/終了/無意志/	Rおわる, Rやむ, Rあがる
〈 終了-有意志 〉	/アスペクト/終了/有意志/	Rおえる, Rあげる
〈 完遂 〉	/アスペクト/完遂/	Rきる, Rぬく, Rつくす, Rとおす, テしまう
〈 場面 〉	/アスペクト/場面/	ところだ, ばかりだ
〈 残存 〉	/アスペクト/残存/	テおく, テある
〈 意志 〉	/モダリティ/表現類型/意志/	テみる, テみせる, つもりだ, つもりがある, つもりもある, 意志形, 省略意志形
〈 希望 〉	/モダリティ/表現類型/希望/	Rたい
〈 願望 〉	/モダリティ/表現類型/願望/	テほしい
〈 依頼 〉	/モダリティ/表現類型/依頼/	テくれるか
〈 命令 〉	/モダリティ/表現類型/命令/	命令形, 文語命令形
〈 疑問 〉	/モダリティ/表現類型/疑問/	か, かい, かしら
〈 推量 〉	/モダリティ/真偽判断/推量/	だろう, 推量形, 省略推量形
〈 可能性 〉	/モダリティ/真偽判断/蓋然性/可能性/	かもしれない, ことがある, こともある
〈 必然性 〉	/モダリティ/真偽判断/蓋然性/必然性/	はずだ, にちがいない
〈 証拠性-観察 〉	/モダリティ/真偽判断/証拠性/観察/	Rそうだ, ようだ, みたいだ
〈 証拠性-伝聞 〉	/モダリティ/真偽判断/証拠性/伝聞/	らしい, そうだ, という, とのことだ
〈 必要 〉	/モダリティ/価値判断/必要/	ざるをえない
〈 適当 〉	/モダリティ/価値判断/適当/	バよい, バいい, タラいい, ほうがいい, べきだ
〈 許容 〉	/モダリティ/価値判断/許容/	テよい, テいい
〈 非許容 〉	/モダリティ/価値判断/非許容/	テハだめだ, タラだめだ, テハならない, タラいけない
〈 説明 〉	/モダリティ/説明/	のだ, ことだ, ものだ, わけだ
〈 丁寧さ 〉	/モダリティ/伝達/丁寧さ/	です, ます, Rなさる, Rくださる, デス列, 謙譲語
〈 態度 〉	/モダリティ/伝達/態度/	かよ, よ, ね, ぜ, ぞ,っけ,っけね,って,ってね, とか, な, なあ, ね, よ, よね, よねえ, わ, わね, わよ
〈 内-授与 〉	/その他/授受/内授与/	テあげる, テやる
〈 他-授与 〉	/その他/授受/他授与/	テくれる
〈 受益 〉	/その他/授受/受益/	テもらう
〈 難易-易 〉	/その他/難易/易しい/	Rやすい, Rよい, Rいい
〈 難易-難 〉	/その他/難易/難しい/	Rがたい, Rにくい, Rづらい
〈 不履行 〉	/その他/不履行/	Rかねる, Rしぶる, Rわすれる, Rそこなう, Rそんじる, Rそびれる
〈 語彙的 〉	/その他/語彙的/	Rすぎる, とする, Rなおす, Rかえす, Rがちだ, Rあう
〈 動詞化 〉	/その他/動詞化/	Rなる, Rする, Rある
〈NONE〉	/その他/NONE/	バならない, バいけない

2. 構成要素の列に対し、意味表現を機械的に合成する。
3. 直前の用言の活用形がもつ意味表現を、文末機能表現の意味表現に追加する。
4. 非構成的意味をもつ構成要素列の意味表現を書き換える。

2.2.1 構成要素に対する意味表現の定義

まず、見出し語リストの合成に用いた 165 種類の構成要素に対し、人手で意味表現を定義した。このうち、120 個の構成要素に対応する意味表現は、意味体系の設計の段階で、意味ラベルとして取り込まれているため、その対応は自明である。

残り 45 個の構成要素に対しては、意味ラベルの列を定義した（表 2)。この表の I は、否定に関わるグループで、たとえば、「ことがない」の意味表現を、〈 可能性=否定 〉と定義した。II は、可能形のグループで、たとえば、「Rだせる」の意味表現を〈 開始=可能 〉と定義した。III は、謙譲語のグループで、たとえば、「テいただく」の意味表現を、〈 受益=丁寧さ 〉と定義した。XI は、接尾辞や終

表 2: 構成要素に定義した意味ラベル列

	意味ラベル列	構成要素の例	数	意味ラベル列	構成要素の例	数
Ⅰ	継続=否定	Rつつない	3	可能性=否定	ことがない	2
	必然性=否定	はずがない	2	説明=否定	わけがない	1
	意志=否定	まい	3			
Ⅱ	開始=可能	Rだせる	1	完遂=可能	Rきれる	5
	終了-無意志=可能	Rおわれる	2	残存=可能	テおける	1
	内-授与=可能	テやれる	1	受益=可能	テもらえる	1
	受益=丁寧さ=可能	テいただける	1	不履行=可能	Rおとせる	1
	語彙的=可能	Rあえる	2			
Ⅲ	内-授与=丁寧さ	テさしあげる	1	他-授与=丁寧さ	テくださる	1
	受益=丁寧さ	テいただく	1			
Ⅸ	受身=否定	(ら) れない	2	受身=必要	(ら) れざるをえない	1
	受身=丁寧さ	(ら) れます	1	使役=否定	(さ) せない	2
	使役=必要	(さ) せざるをえない	1	使役=丁寧さ	(さ) せます	1
	使役=受身=否定	(さ) せ (ら) れない	2	使役=受身=丁寧さ	(さ) せ (ら) れます	1
	使役=受身	(さ) せ (ら) れる	1	使役=受身=必要	(さ) せ (ら) れざるをえない	1
	疑問=態度	かね	3			

助詞の意味表現からなるグループで、たとえば、「(ら) れます」の意味表現を、⟨ 受身=丁寧さ ⟩ と定義した。

2.2.2 構成要素の列に対する意味ラベルの合成

先に述べたように、シソーラスの見出し語は、構成要素を接続することによって生成される。この構成要素の接続と同時に、意味表現の合成を行う。以下に、構成要素 L の直後に構成要素 R を接続する手順を示す。なお、活用形を k_i、構成要素または活用形 X の意味表現を $s(X)$ と表す。

1. 辞書形として与えられる L に対し、可能な活用形 k_i に対応した L_i を作成する。この際、L_i の意味表現 $s(L_i)$ を、$s(L_i) = s(L) + s(k_i)$ により合成する。
2. L_i の直後に R を接続できる場合、それらを接続した列 L_iR を生成する。この際、L_iR の意味表現を、$s(L_iR) = s(L_i) + s(R)$ により合成する。

なお、この手順で、構成要素の接続性は、L_i が R に対して定義される左接続条件[3]を満たし、かつ、L_iR のいずれかの表記が『現代日本語書き言葉均衡コーパス』に出現する場合に、接続できると判定する。

例として、「テいる (L)」の直後に「のだ (R)」を接続する場合を考える。手順 1 で、L に対して、終止形「テいる (L_1)」、テ形「テいて (L_2)」、タ形「テいた (L_3)」などを生成する。ここで、終止形やテ形は、対応する意味表現をもたないので、$s(k_i)$ は空となり、$s(L_1) = s(L_2) = s(L) =$⟨ 継続 ⟩となる。一方、タ形は、意味表現 $s(k_3) =$⟨ 過去 ⟩をもつので、$s(L_3) =$⟨ 継続=過去 ⟩となる。この後、直後に R(「のだ」) が接続可能な L_1 と L_3 に対して手順 2 が適用され、「テいるのだ (L_1R)」と「テいたのだ (L_3R)」が生成される。この際、$s(L_1R) =$⟨ 継続=説明 ⟩、$s(L_3R) =$⟨ 継続=過去=説明 ⟩が合成される。

見出し語生成は、このような構成要素の接続を繰り返すことによって実現される。すなわち、まず、構成要素 A と構成要素 B を接続することで、見出し語 AB を生成する。次に、こうして生成した見出し語 AB を新たな構成要素とし、さらに構成要素 C を接続することで、見出し語 ABC を生成する。このようにして、より長い文末機能表現が生成される。

[3]各構成要素には、その構成要素がどのような構成要素に接続可能であるかの制約が記述されている。これを、左接続条件と呼ぶ。

2.2.3 直前の用言の活用形がもつ意味表現

すでに述べたように、活用形のいくつかは、それ自身が意味をもつ。たとえば、「書こう（意志形）」は、⟨ 意志 ⟩という意味をもつ。一方、文末機能表現「つもりだ」は、やはり、⟨ 意志 ⟩という意味をもつ。しかしながら、「書こう」は、明示的な文末機能表現をもたないため、このままでは、「書こう」を「書くつもりだ」に正規化することができない。

この問題に解決するために、本シソーラスでは、文末用言の活用形を、表層形をもたない擬似的な文末機能表現とみなし、それに相当するエントリを設定する方法を採用する。具体的には、意味をもつ活用形 k それぞれに対し、表記が空文字列（λ）で、意味表現を $s(k)$ とするエントリを設定する。このエントリは、活用形 k にのみ接続可能とする。このようなエントリを設定することにより、たとえば、文末用言「書こう」は、用言「書く」の意志形に、意味表現⟨ 意志 ⟩をもつ表記 λ の文末機能表現が接続していると解釈することが可能となり、その結果、「書こう」を「書くつもりだ」に正規化することが可能となる。

上記で述べた擬似的な文末機能表現の導入と整合させるために、シソーラスの各エントリには、直前用言の活用形 k の意味情報を取り込む。具体的には、2.2.2 節で生成されたエントリの意味表現の左側に、直前用言の活用形の意味情報を追加する。たとえば、エントリ「のだ（⟨ 説明 ⟩）」に対して、タ形（⟨ 過去 ⟩）に接続する場合の意味表現⟨ 説明=過去 ⟩を合成する。この結果、 1 つのエントリは、直前用言の活用形に依存した、複数の意味表現をもつこととなる。

2.2.4 意味表現の書き換え

文末機能表現が非合成的意味をもつ場合、2.2.2 節や 2.2.3 節で述べた意味表現の合成法では、適切な意味表現を生成することができない。たとえば、「テもらいたい」の意味表現は、その構成要素から⟨ 受益=希望 ⟩と合成されるが、適切な意味表現は⟨ 願望 ⟩である。このような、意味の非合成性に対処するために、合成して得られた意味表現に書き換えルールを適用し、適切な意味表現へと変換する。現在までに、定義した書き換えルールを表 3 に示す。

この表に示すように、書き換えルールは、大きく 2 種類に分類される。I は、特定の文末機能表現に対するルール群である。たとえば、ルール I(1) は、文末機能表現「R なさる」の命令形「R なさい」の意味表現を⟨ 命令 ⟩へと書き換えるためのルールである。これに対して、II は、意味ラベルの上位レベルのみを参照するルール群である。たとえば、ルール II(3) は、⟨ 否定 ⟩の直後に⟨ 疑問 ⟩がくる場合、この部分を⟨ 疑問 ⟩へと書き換える。このルールによって、「ないかしら」「テないか」などの意味表現が⟨ 疑問 ⟩に書き換えられる。

書き換えルールの適用は、以下の手順で行う。ここで、書き換え対象の意味表現を $S = s_n \cdots s_0$、書き換え後の意味表現を T と表し、ルールは、k 個の意味ラベルの列を、l 個の意味ラベルの列に書き換えるものとする。

1. $i = 0$ とする。
2. 書き換え対象の意味表現 $s_{i+k-1} \cdots s_i (0 \leq i \leq n,\ 1 \leq k)$ に適用可能なルールをすべて求める。
3. 適用可能なルールが存在した場合、コストが最小のルールを適用し、書き換え後の意味表現を T の先頭に追加する。$i = i + k$ とする。
4. 適用可能なルールが存在しなかった場合、s_i を T の先頭に追加し、$i = i + 1$ とする。
5. $i \leq n$ の場合、手順 2 に戻る。

たとえば、構成要素から合成される「テおいてくれるか」の意味表現は、⟨ 残存=他-授与=疑問 ⟩である。$i = 0$ で、ルール II(1) が適用可能であり、⟨ 他-授与=疑問 ⟩が⟨ 依頼 ⟩に書き換えられる。この結果、「テおいてくれるか」の意味表現は、⟨ 残存=依頼 ⟩となる。

表 3: 現在までに定義した書き換えルール

		書き換え前	書き換え後	コスト	適用する表現の例
I	(1)	*/Rなさる = */命令形	⟨ 命令 ⟩	3	Rなさい
	(2)	*/テやる = */謙譲語 = */命令形	⟨ 依頼 ⟩	3	テください
	(3)	*/Rかねる = */ない	⟨ 可能性 ⟩	4	Rかねない
	(4)	説明/* = */Rする	⟨ 意志 ⟩	4	ようにする
	(5)	否定/* = */バならない	⟨ 必要 ⟩	4	なければならない
		否定/* = */バいけない	⟨ 必要 ⟩	4	なければいけない
	(6)	意志/* = */とする	⟨ 意志 ⟩	4	ウとする
II	(1)	他-授与/* = 疑問/*	⟨ 依頼 ⟩	4	テくれるか
		他-授与/* = 丁寧さ/* = 疑問/*	⟨ 依頼=丁寧さ ⟩	4	テくださるか
		受益/* = 可能/* = 疑問/*	⟨ 依頼 ⟩	4	もらえるか
		受益/* = 丁寧さ/* = 可能/* = 疑問/*	⟨ 依頼=丁寧さ ⟩	4	テいただけるか
	(2)	受益/* = 希望/*	⟨ 願望 ⟩	4	テもらいたい
		受益/* = 丁寧さ/* = 希望/*	⟨ 願望=丁寧さ ⟩	4	テいただきたい
	(3)	否定/* = 疑問/*	⟨ 疑問 ⟩	6	ないか
	(4)	意志/* = 意志/*	⟨ 意志 ⟩	6	テみよう (「テみる」の意志形)
		推量/* = 推量/*	⟨ 推量 ⟩	6	でしょう (「だろう」の推量形)

注意：ルール中の記号 '*' は、任意の上位クラス、あるいは、最下位クラスにマッチすることを表す。

表 4: 作成したシソーラスのエントリの例

見出し /準用言/Rはじめる–/連用形/テ形=/準用言/ている–/辞書形	**ID** X010=X510
表記 はじめている, 始めている	**頻度** 364,559
活用型 /動詞/母音動詞	**活用形** /辞書形
左接続条件 /準用言/連用形	**意味表現** ⟨ 開始=継続 ⟩
見出し /助動詞/のだ–/デス列/終止形=/終助詞/か–/辞書形	**ID** Y200=Z010
表記 のですか, んですか	**頻度** 13996,15076
活用型 /無活用型	**活用形** /辞書形
左接続条件 /助動詞/のだ/タ形接続; /助動詞/のだ/基本形接続	**意味表現** ⟨ 過去=説明=丁寧さ=疑問 ⟩; ⟨ 説明=丁寧さ=疑問 ⟩
見出し /用言活用形/意志形	**ID** P020
表記 λ	**頻度** 1
活用型 –	**活用形** –
左接続条件 /用言活用形/意志形接続	**意味表現** ⟨ 意志 ⟩

2.3 作成したシソーラス

以上の手順で、45,948 エントリからなるシソーラスを作成した。エントリの例を表 4 に示す。この表に示すように、1 つのエントリは**見出し**、**ID**、**表記**、**頻度**、**活用型**、**活用形**、**左接続条件**、**意味表現**の 8 つの要素からなる。

これらの要素のうち、**見出し**と **ID** は、そのエントリに固有な識別子である。**表記**は、そのエントリの出現形を示す。出現形が複数存在する場合は、それらが列挙される。**頻度**は、表記に記述された出現形が、『現代日本語書き言葉均衡コーパス』に出現する回数を示す。**活用型**と**活用形**は、そのエントリの活用情報を表す。

左接続条件は、そのエントリが接続可能な用言に対する制約情報を表現している。具体的には、接続可能な用言の集合を規定するラベル（左接続キー）を列挙する。たとえば、助動詞「のだ」に対して使用される左接続キー "/助動詞/のだ/タ形接続" は、動詞・形容詞のタ形に接続可能であることを示す。

意味表現には、2.2.3 節で述べたように、直前用言の活用形に依存した複数の意味ラベル列が列挙される。これらは、左接続条件で列挙される左接続キーに対応する。

2.4 活用形展開シソーラス

前節で述べたシソーラスのエントリのうち、活用するエントリの活用形は、すべて辞書形である。それに対して、実際の文の文末は、タ形や意志形などの活用形をとりうる。これに対応するために、

先に述べたシソーラスから、文末になりうるすべての活用形を生成（展開）したシソーラスを自動生成する。この結果、45,948 エントリのシソーラスから、140,233 エントリの活用形展開シソーラスが得られた。

3　正規化辞書の生成

述部正規化システムは、前節で述べた活用形展開シソーラスをそのまま使用するのではなく、ユーザーの正規化定義に基づいて意味表現を書き換えることによって生成される**正規化辞書**を使用する。正規化辞書生成のための意味表現の書き換えは、２段階に分けて行う。なお、述部正規化システムでは、代表表現は頻度に基づいて自動的に決定するため、正規化辞書では、代表表現を明示的には定義しない。

3.1　意味ラベル単体に対する整理・集約

第１段階の書き換えでは、意味ラベルの整理・集約を行う。ユーザーは、文末機能表現シソーラスのそれぞれの意味ラベルに対して、どのような書き換えを行うかを定義する。システムは、この定義に従って、シソーラスのエントリの意味表現を書き換える。

意味ラベルの書き換えには、次の３種類がある。

1. シソーラスの意味ラベルをそのまま使用する（書き換えない）。
2. 意味ラベルを削除する。（つまり、ある種の意味情報を無視する）
3. 新たな意味ラベルへと書き換える。（つまり、シソーラスが採用している意味体系に含まれない、独自の意味区分を導入する。）

シソーラスの意味ラベルの種類は、リテラルや活用形に対応する最下位レベルを除くと、39 種類である。この粒度で正規化したいのであれば、意味ラベルを書き換える必要はない。一方、より大きな粒度で正規化を行いたい場合は、特定の意味ラベルを削除するか、あるいは、複数の意味ラベルを統合する新しい意味ラベルを導入することで、これを実現する。意味ラベルの書き換えを指示するユーザー定義の例を、表５に示す。

ユーザー定義は、書き換えるべき意味ラベルの集合と、それに対する書き換えコマンドで構成される。意味ラベルの集合は、根ノードからのパス表現（あるいは、最下位レベルを省略した略記表現）で記述する。たとえば、定義 (7) のパス表現⟨/モダリティ/真偽判断/⟩は、このパスと前方一致する意味ラベルの集合、すなわち、⟨ 推量 ⟩、⟨ 可能性 ⟩、⟨ 必然性 ⟩、⟨ 証拠性-観察 ⟩、⟨ 証拠性-伝聞 ⟩からなる集合を意味する。

書き換えコマンドには、次の３種類がある。コマンド ‘+’ は、意味ラベルを書き換えないことを指示する。コマンド ‘−’ は、書き換え対象の意味ラベルを削除することを指示する。‘+’ と ‘−’ 以外のコマンドは、書き換え対象の意味ラベルを、そのコマンド（意味ラベル）へと書き換えることを指示する。たとえば、定義 (3) は、⟨ 説明 ⟩を削除することを指示する。この定義に基づき、「テいるのだ」の意味表現⟨ 継続=説明 ⟩が⟨ 継続 ⟩に書き換えられる。一方、定義 (5) と (6) は、それぞれ⟨ 願望 ⟩と⟨ 依頼 ⟩を⟨ 要求 ⟩に書き換えることを指示する。これらの定義に基づき、「テほしい⟨ 願望 ⟩」や「テくれるか⟨ 依頼 ⟩」が、いずれも⟨ 要求 ⟩に書き換えられる。

3.2　意味ラベル列に対する書き換え

第２段階の書き換えでは、整理・集約後の意味ラベルの列に対して、さらなる書き換えを行う。先の第１段階の書き換えは、意味ラベル単体に対して定義される。これに対して、第２段階の書き換えでは、隣接する意味ラベルを考慮して、書き換えるか否かを指定することができる。表６に、ユーザーの定義例と適用例を示す。

この表に示すように、ユーザーは、書き換え前の意味ラベル列、書き換え後の意味ラベル列、コス

表 5: 意味ラベル単体に対する定義の例

	もとの意味ラベル	コマンド		もとの意味ラベル	コマンド
(1)	⟨ 開始 ⟩	+	(2)	⟨ 継続 ⟩	+
(3)	⟨ 説明 ⟩	−	(4)	⟨/モダリティ/伝達/*⟩	−
(5)	⟨ 願望 ⟩	要求	(6)	⟨ 依頼 ⟩	要求
(7)	⟨/モダリティ/真偽判断/⟩	推測			

表 6: 意味ラベル列に対する定義の例

	書き換え前	書き換え後	コスト	適用例
(1)	⟨ 開始 = 意志 ⟩	⟨ 意志 ⟩	4	Rはじめよう
(2)	⟨ 終了 = 過去 ⟩	⟨ 過去 ⟩	4	テしまった
(3)	⟨ 開始 = 継続 ⟩	⟨ 継続 ⟩	6	Rはじめている
(4)	⟨ 継続 = 過去 ⟩	⟨ 過去 ⟩	6	テいた

トの３つ組を記述する。たとえば、定義 (3) は、⟨ 開始=継続 ⟩を⟨ 継続 ⟩に書き換えることを指示している。システムは、2.2.4 節と同様の方法で、この定義に基づき、意味ラベル列を書き換える。たとえば、定義 (3) に基づき、「Rはじめている」の意味表現⟨ 開始=継続 ⟩が⟨ 継続 ⟩に書き換えられる。その結果、「Rはじめている」は、「テいる⟨ 継続 ⟩」と同じグループに分類されることになる。

4 述部正規化システム

述部正規化システムは、前節で述べた正規化辞書に基づいて、文末機能表現を同定し、正規化する。本システムの構成を図 2 に示す。この図に示すように、本システムは、**文末機能表現同定システム**と**言い換えシステム**の２つのサブシステムから構成される。

4.1 文末機能表現同定システム

文末機能表現同定システムは、形態素解析済の文を受けとり、その文の文末機能表現を同定し、意味表現（意味ラベル列）を付与する。入力形態素列を $m_n \cdots m_0$(添字は文末に近いほど小さい)、正規化辞書のエントリを e とするとき、本システムは、以下の２つの条件を満たす形態素列 $m_i \cdots m_0 (-1 \leq i < n)$ のうち最大の i をとる形態素列[4]を、文末機能表現 e と同定する。

条件 1 形態素列 $m_i \cdots m_0$ の表記が、正規化辞書のエントリ e の表記と一致する。
条件 2 形態素 m_{i+1} は、エントリ e の左接続条件（左接続キー）を満たす。

なお、このような条件を満たす形態素列 $m_i \cdots m_0$ が存在しなかった場合は、入力文中に文末機能表現は存在しないと判定する。文末機能表現が同定された場合は、その形態素列に、対応するエントリ e の、条件を満たした左接続キーに対する意味表現を付与して出力する。

たとえば、入力形態素列「本/を/書き/始めて/いる/に/ちがいない」に対し、上記の２つの条件を満たすエントリとして、「にちがいない」、「テいるにちがいない」、「始めているにちがいない」の３つが見つかる。これらのうち、最も長い（最大の i をとる）「始めているにちがいない」を、文末機能表現と同定し、その意味表現⟨ 継続=推測 ⟩を付加して出力する。

4.2 文末機能表現言い換えシステム

入力文中に文末機能表現が同定された場合、それを代表表現に書き換えて出力する。この手順を以下に示す。なお、意味表現 s に対して、s に含まれるそれぞれの意味ラベルの最下位レベルを除いた

[4] $i = -1$ の場合は、空文字列 λ とする。

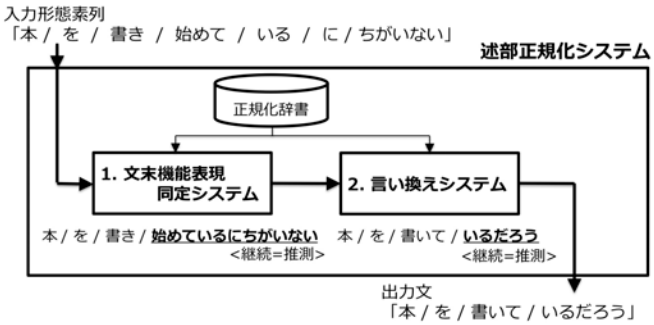

図 2: 述部正規化システムの全体像

ものを $g(s)$ と表す。

1. 代表表現の決定
   同定された文末機能表現 e に付与された意味表現 $s(e)$ とし、e の直前の形態素を m とするとき、次の 2 つの条件を満たす正規化辞書エントリ e' のうち、頻度が最も高い表記をもつエントリを代表表現 e_r とする。

   条件 1 エントリ e' のある左接続キーに対応する意味表現を $s(e')$ とするとき、$g(s(e')) = g(s(e))$ である。

   条件 2 用言 m は、条件 1 の左接続キーが要求する活用形をとりうる。

2. 代表表現への置換
   文末機能表現 e を代表表現 e_r に置き換える。このとき、必要があれば、用言 m の活用形を、e_r が接続できる形に変更する。

先に示したように、入力形態素列「本/を/書き/始めて/いる/に/ちがいない」に対して、「始めているにちがいない⟨ 継続=推測 ⟩」が同定される。意味表現⟨ 継続=推測 ⟩をもち、かつ、用言「書く」に接続可能なエントリとして、「R つつあるはずだ」「テいるだろう」などがみつかり、これらのうち、頻度が最も高い「テいるだろう」が代表表現として選ばれる。最終的に、「書く」の活用形が連用形からテ形に変更され、代表表現を接続した「本を書いているだろう」が生成される。

4.3 正規化の例

以下に、作例に対するシステムの出力を示す。なお、(1) には文末機能表現同定システムの出力を、(2) には言い換えシステムの出力を示し、同定された、あるいは、言い換えによって得られた文末機能表現を、下線部で示す。

例 1 「窓 / を / 開けて / いただき / たい / の / です」
(1) 窓 / を / 開けて / いただきたいのです⟨ 要求 ⟩
(2) 窓 / を / 開けて / ほしい⟨ 要求 ⟩

例 2 「対立 / 関係 / が / 生じ / かね / ない / の / かも / しれ / ない」
(1) 対立 / 関係 / が / 生じ / かねないのかもしれない⟨ 可能性 ⟩
(2) 対立 / 関係 / が / 生じる / かもしれない⟨ 可能性 ⟩

例 3 「席 / を / 譲って / くれ / ない / かしら」
(1) 席 / で / 譲って / くれないかしら⟨ 要求 ⟩
(2) 席 / で / 譲って / くれるか⟨ 要求 ⟩

5 おわりに

本論文では、文末機能表現シソーラスと、それを利用した述部正規化システムについて述べた。現時点の状況は、全体の枠組みがほぼ完成したところである。今後、意味体系の見直しや、シソーラスの意味表現書き換えルールを整備することにより、シソーラスをより良いものとすると同時に、述部正規化システムの性能向上を目指す予定である。

謝辞 本研究は，JSPS 科学研究費基盤研究 (B)「平易な日本語表現への工学的アプローチ」課題番号 24300052 の助成を受けている。本研究では，現代日本語書き言葉均衡コーパス DVD 版を利用した。

参考文献

[1] T. Izumi, K. Imamura, G. Kikui, and S. Sato. Standardizing Complex Functional Expressions in Japanese Predicates: Applying Theoretically-Based Paraphrasing Rules. In *23rd International Conference on Computational Linguistics*, p. 64, 2010.

[2] 日本語記述文法研究会（編）. 現代日本語文法 4 第 8 部 モダリティ. くろしお出版, 2003.

[3] 日本語記述文法研究会（編）. 現代日本語文法 3 第 5 部 アスペクト 第 6 部 テンス 第 7 部 肯否. くろしお出版, 2007.

[4] 益岡隆志. 日本語モダリティ探究. くろしお出版, 2007.

[5] 松木久幸, 佐藤理史, 駒谷和範. 文末機能表現シソーラスの編纂に向けて–文末機能表現の網羅的生成–. 言語処理学会 第 18 回年次大会 発表論文集, 2012.