

字面解析を応用した日本語文章推敲支援ツールの開発

菅沼 明 牛島 和夫

九州大学 大学院システム情報科学研究科

我々は、接続助詞「が」や否定表現、受身形などの候補を字面だけの情報で抽出する方法を構築し、それを活用して日本語文章推敲支援ツール『推敲』を開発している。本稿では、まず、簡単な表を利用した字面解析手法を用いて日本語文章中にある活用語の活用形を推定する方法（「活用チェック法」と呼ぶ）について述べる。活用チェック法を用いることで、再現率を100%に保ったまま、適合率を約10%向上させることができた。次に、本稿では『推敲』のワークステーション(WS)上での実現についても述べている。WS版『推敲』を設計する際に解析部とインタフェースとを分離し、ユーザインタフェースとしてmuleを使用することで、エディタとの連携を可能とした。

Development of a System of Writing Tools SUIKOU with a Textual Analysis for Japanese Documents

Akira SUGANUMA and Kazuo USHIJIMA

Graduate School of Information Science and Electrical Engineering
Kyushu University

6-1 Kasuga-koen, Kasuga-shi, Fukuoka-ken, 816 Japan

e-mail : suga@is.kyushu-u.ac.jp

ushijima@csce.kyushu-u.ac.jp

We constructed an extraction method with only textual information for the passive voice, the negative expressions, the conjunctive particle "GA," and so on. And we are developing a system of writing tools SUIKOU with the textual analysis method. This paper describes a method to estimate the conjugation of Japanese verbs and adjectives with a textual analysis. The precision of a new extraction method with the estimating method is improved by about 10% in comparison with that of old one. In this paper, we describe about implementation of SUIKOU on a workstation too. SUIKOU on WS is designed with separation of an analysis engine and a user interface. By using "mule" for the user interface, SUIKOU can cooperate with an editor.

1. はじめに

我々の研究室では、日本語文章推敲支援ツール『推敲』の開発を行っている^[1, 2]。このツールは、機械可読な日本語文章を字面だけで解析して、推敲に役立つ情報を書き手に提供するものである。このツールは次の2つの方針のもとで開発を進めてきた。

- (1) 文章中に問題となりそうな箇所があればそれを指摘できればよい。(実際に推敲するのは書き手である)
- (2) 実用規模(1万字程度)の文章を待ち遠しくない時間で処理して欲しい。ユーザは、コマンドを入力して10秒も待たされると待ち遠しいと感じる。

我々は『推敲』に使用するために、推敲に役立つ情報を字面解析だけで抽出する方法を構築してきた^[3, 4, 5]。それらの抽出法は、実際の日本語文章を調査し、その結果を参考にして構築してきた。字面解析はオンメモリでの処理が可能なので、処理能力が小さい計算機であっても高速な解析が可能である。また、処理対象の文章が大規模になっても、大幅に待たされることなく処理が終了することが期待できる。

2. 字面解析手法

『推敲』には指示詞、受身、接続助詞「が」、否定表現、とりたて詞(副助詞、係助詞の一部)のような文法的な意味を持つ単語を指摘する機能がある。それらを文章中から取り出すために、個々に抽出法を構築してきた。それらは、文字列照合を基本としているが、照合の後に前後の文字に関するいくつかの判定条件を付加しているものもある。例えば、表1に示す判定条件を満たすものを接続助詞「が」として抽出する。

『推敲』では辞書を使わずに字面だけの情報で抽出を行っているため、抽出精度は文法処理を行った場合よりも悪くなることが予想できる。この抽出精度の指標として、情報検索の分野で使用されている再現率と適合率を使用する。再現率と適合率は以下の式で定義される。

$$\begin{aligned} \text{再現率} &= \frac{\text{候補中に含まれる抽出すべき対象の数}}{\text{文章中の抽出すべき対象の数}} \times 100(\%) \\ \text{適合率} &= \frac{\text{候補中に含まれる抽出すべき対象の数}}{\text{抽出法で得られる候補の数}} \times 100(\%) \end{aligned}$$

表 1: 接続助詞「が」の抽出法

判定条件 1	「が」が接続助詞であるためには、その1文字前が「う、く、す、つ、ぬ、む、る、ぐ、ぶ、い、だ、た、ん」のいずれでなければならない。
判定条件 2	「が」の1文字後が促音、撥音である場合、その「が」は接続助詞でない。
判定条件 3	「が」の1文字前が「だ」であるとき、その「だ」が文頭の文字であれば、その「が」は接続助詞でない。
判定条件 4	「が」の1文字前が「う」であるとき、その「う」の1文字前が「ほ」であれば、その「が」は接続助詞でない。
判定条件 5	「が」の1文字前が「つ」であるとき、その「つ」の1文字前が数字または漢数字であれば、その「が」は接続助詞でない。

文章から問題となる箇所を抽出する際に犯す誤りには2種類ある。1つは第一種の誤り「指摘すべきものを取りこぼしてしまう」で、もう1つは第二種の誤り「指摘すべきでないものまで指摘してしまう」である。これら2つの誤りは再現率、適合率と密接な関係がある。抽出の際に第一種の誤りを犯せば、再現率が下がり、第二種の誤りを犯せば、適合率が下がる。『推敲』に組み込んでいる字面解析手法は、第一種の誤りを犯さないで、再現率は100%である。また、『推敲』の開発方針(1)から、第二種の誤りはある程度許容している。そのため、抽出するものにもよるが、適合率は約90%である。しかし、第二種の誤りも少なければ少ないほど良いので、再現率100%のもとで適合率をできるだけ高くするような抽出法の構築を行ってきた。

3. 活用チェック法

字面解析手法の構築をとおして、文全体に渡る解析を行わなくても、文中に存在する用言の位置と活用形の情報がわかれば、推敲に役立つ情報の抽出が可能であることがわかった。活用チェック法は、字面の情報から用言の活用形を推定するものである^[6]。活用チェック法を使用することで、これまで『推敲』で使用していた字面解析手法と比べて抽出精度を向上させることができる。

活用チェック法では、用言を語幹と活用語尾に分け、それぞれで品詞を推定する。語幹と活用語尾とで推定した品詞が一致すれば、該当の

表 2: 漢字で終る語幹の分類

語幹の品詞	漢字数	割合
動詞のみ	1,204	40.6%
形容詞のみ	11	0.3%
形容動詞のみ	142	4.8%
動詞, 形容詞	18	0.6%
動詞, 形容動詞	417	14.1%
形容詞, 形容動詞	15	0.5%
動詞, 形容詞, 形容動詞	52	1.8%
語幹の最後にはならない	1,106	37.3%
合計	2,965	—

品詞・活用形で用言が存在すると判断する。語幹と活用語尾とに分けて処理するために、用言をそれぞれ2つの部分に分けて表に登録する。

3.1 語幹表

昭和56年に出された内閣告示第三号「送り仮名の付け方」によると、送り仮名は活用語尾を送るのが通則とされている。そのため、用言を漢字仮名混じりで書く場合、語幹の最後の文字は漢字であることが多いと考えられる(例えば、動詞「見分ける」の場合、「見分」が語幹であるので文字「分」が語幹の最後の文字となる)。

漢字によって動詞として使われるもの、形容詞として使われるものなど、使われ方に特徴があると考えると、漢字で終わる語幹について品詞の推定を行うことが可能である。用言のうち語幹が漢字で終わる単語を公用データベース日本語単語辞書⁷⁾を用いて調査した。JIS第一水準漢字2,965字に関して用言のどの品詞として使われるかを調べると、表2のようになる。

表2から、1,357種類の漢字(45.7%)は、動詞か形容詞、形容動詞のいずれかの語幹の最後であることが一意に定まることが分かる。また、1,106種類の漢字(37.3%)は用言の語幹の最後の文字にはならない。つまり、83%の漢字は、漢字だけの情報で語幹の品詞(動詞, 形容詞, 形容動詞または用言にはならない)を推定できることになる。この調査結果を基にして、漢字と品詞を組にして表を作成した(大きさ約12KB)。この表を漢字表と呼ぶ。品詞は、動詞, 形容詞, 形容動詞を登録し、動詞はさらに五段活用(活用の行も区別), 上一段活用, 下一段活用, カ変, サ変, ザ変の活用型も区別して登録した。

語幹の最後の文字が平仮名である用言と用言が平仮名書きされた場合とを処理するために、

平仮名をキーとして引く平仮名表を漢字表と同様にして作成した(大きさ約0.4KB)。品詞を登録する際に、平仮名書きされた用言も登録しているため、1つの平仮名文字に対して登録している品詞の数が多くなっている。そのため、平仮名表は、品詞の候補を絞るのに漢字表ほどは有効ではない。しかし、用言の語幹の最後の文字になり得ない平仮名も明かに存在するので、平仮名表を用意して語幹の品詞の判定に用いる。

3.2 活用語尾表

活用語尾の処理では、文字列から品詞と活用形を求めたい。また、助動詞の活用形も用言の活用語尾と同様の処理が必要である。そのため、用言の活用語尾と助動詞の活用形とを登録した表を作成し(この節では、助動詞の活用形と用言の活用語尾の両方をあわせて活用語尾と記す)、活用語尾表と呼ぶ(全体で大きさ約19.6KB)。活用語尾表は図1の構造を持っており、活用語尾テーブルと活用語尾ツリーからなる。

● 活用語尾テーブル

図1にあるように、活用語尾テーブルは文字と活用形とでインデックスをつけた表で、活用語尾の最後の文字からその文字列がどのような活用形になることが可能かが引けるようになっている。もし、ある文字がある活用形となることが可能であれば、そのノードには活用語尾ツリーへのリンクが登録されている。このテーブルを参照してNULLが登録されていれば、その文字で終わるその活用形はないことを表している。

● 活用語尾ツリー

活用語尾ツリーはTRIE構造をしており、活用語尾の最後の文字以外を登録している。このツリーのノードには、そのノードに到達した場合にどんな品詞の活用語尾になり得るかが登録してある。また、文字列を後ろから前に遡る方向でツリーを作成している。そのため、ある文字から文頭の方向に遡った文字列が活用語尾ツリーに登録してあれば、その文字列の品詞と活用形を判断できる。

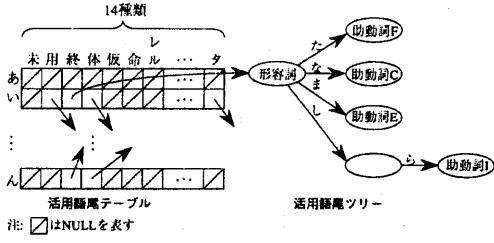


図 1: 活用語尾表

3.3 表を使用した品詞の推定

3.1節, 3.2節で述べた3つの表(漢字表, 平仮名表, 活用語尾表)を使用して, 以下の手順で品詞と活用形の推定を行う。この処理を始めるに当たって推定を始める文字の位置と要求する活用形をパラメータとして与える。

- (1) 推定開始位置の文字と要求された活用形とで活用語尾テーブルを参照し, 活用語尾ツリーへのリンクがあるか否かを調べる。もし, 活用語尾テーブルに NULL が登録されていれば, 推定開始位置より前側の文字列は要求された活用形ではないとして推定を終了する。
- (2) 推定開始位置からテキストを遡ってスキャンし, 文章中の文字列が活用語尾ツリーに登録されているか否かを調べる。文字列が登録されていれば, 活用語尾ツリーに登録してある品詞を一時的に保持しておく。登録されていない場合には活用語尾ではないとして推定を終了する。
- (3) 手順(2)で活用語尾とした文字列の1文字前の文字を調べ, 漢字である場合は漢字表を, 平仮名である場合は平仮名表を引く。
- (4) 漢字表もしくは平仮名表に登録されている品詞と, 手順(2)で一時的に保持した品詞とが一致した場合には品詞と活用形を確定させて, 推定を終了する。一致しない場合は活用語尾ではないとして推定を終了する。

図2は, 接続助詞「が」の抽出を例として活用チェック法の動作を表わしたものである。文章中にある文字「が」が接続助詞であるか否かを調べるために, 文字「が」の1文字前の文字

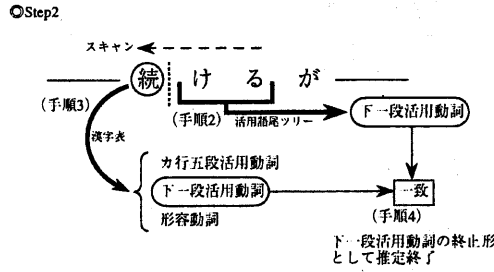
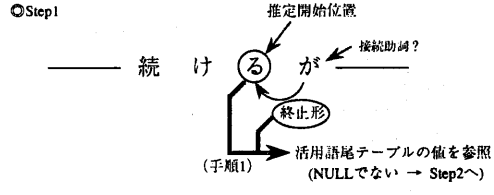


図 2: 活用チェック法の動作

「る」を推定開始位置として活用チェック法を始める。その際, 接続助詞「が」は活用語の終止形に接続するので, 要求する活用形は終止形となる。上の手順に従って処理を行うと, 文字「る」より前側の文字列は下一段動詞の終止形として推定して終了する。

4. 接続助詞「が」の抽出への適用

接続助詞「が」は, 順接, 逆接, ただ2つの句をつなぐだけ, という3つの用法を持っており, どのような関係にある2つの句でも接続することができる。この性質により, 接続助詞「が」を文章中に用いると書き手の意図が読み手に正しく伝わらないことが起こりうるため, 文章を作成する際に接続助詞「が」の使用を吟味したほうがよい。

4.1 判定条件

従来法では, 表1に示した5つの判定条件によって接続助詞「が」の抽出を行っている。これらの条件のうち判定条件2以外の4つは文字「が」の前にある文字列が活用語の終止形になり得るかを判定する条件である。そのため, この4つの条件は, 活用チェック法による活用語の終止形の推定に置き換えることができる。

4.2 抽出精度の比較

活用チェック法による接続助詞「が」の抽出法を使用して, 実際の文章中から接続助詞「が」を抽出した。調査対象の文章は我々の研究室で

表 3: 接続助詞「が」の抽出結果

調査対象文章：67 万字文章			
項 目	数	適合率	再現率
「が」の総数	6,987	—	—
従来法で抽出する候補	437	91.3%	100%
活用チェック法で抽出する候補	407	98.0%	100%
接続助詞「が」	399	—	—

調査対象文章：200 万字文章			
項 目	数	適合率	再現率
「が」の総数	30,570	—	—
従来法で抽出する候補	3,799	85.2%	100%
活用チェック法で抽出する候補	3,377	95.8%	100%
接続助詞「が」	3,236	—	—

蓄えている以下の機械可読な日本語文章である。

67 万字文章: 研究室で書かれた科学技術論文 (総文字数 669,842 文字)。この文章を調査した結果を用いて表 1 に示した判定条件を構築した。

200 万字文章: 朝日新聞記事データ (総文字数 1,981,950 文字)。1988 年版, 1 月～6 月から抜粋。

調査結果を表 3 に示す。67 万字文章における調査結果によると、活用チェック法を利用した抽出法で抽出した場合、従来法で抽出した場合よりも第二種の誤りが 30 個減っている。また、再現率は 100% のままである。また、従来法でふるい落としした接続助詞でない「が」は、活用チェック法を使用した抽出法でもすべて候補からふるい落とししている。抽出の結果、適合率は 91.3% から 98.0% になり、約 7% 向上している。従来法で抽出した候補の中には、ワ行五段動詞の転生名詞に格助詞「が」が接続したもの (例えば「違いが」) が 17 個含まれていた。この第二種の誤りを活用チェック法で全て取り除くことができる。このことが適合率向上の大きな要因として挙げられる。

調査対象文章を 200 万字文章とした場合でも、表 3 に示すように、従来法と比べて適合率が 10% 向上している。また、再現率は 100% を保っている。

4.3 応答時間

ユーザがキーボードからコマンドを発すると、『推敲』はすべての候補を検索し、結果の先頭部分を画面に表示する。画面からあふれた結果に対して、ユーザは画面をスクロールさせて候

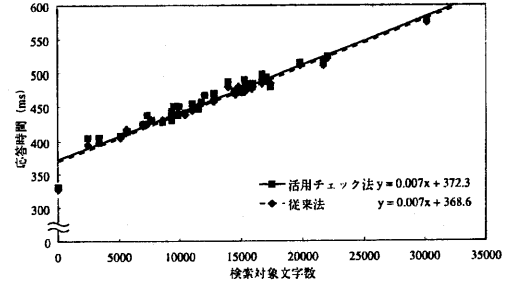


図 3: 接続助詞「が」抽出の応答時間の比較

補を 1 つ 1 つ吟味していく。このように、『推敲』を使用する際には、コマンドを発してから先頭部分の画面が表示されるまでの時間 (応答時間) が問題となる。

応答時間の測定にあたって、従来法による接続助詞「が」の抽出法と活用チェック法による抽出法とをパソコン (MS-DOS) 版『推敲』Ver.1.9 のコマンド “接続助詞「が」の抽出” に実装した。このコマンドを実行してから、結果が画面に出力されるまでの時間を測定した。測定には PC-9801 RA (CPU 80386 CLOCK 16MHz) を使用し、調査対象文章は、研究室で蓄えている科学技術文献である。測定結果を図 3 に示す。

応答時間は 1 万字の文章では 0.45 秒、3 万字の文章でも 0.6 秒程度であった。また、従来法も活用チェック法もほとんど変わらない応答時間を示した。これは、推敲の開発方針である「実用規模 (1 万字程度) の文章を待ち遠しくない時間で処理して欲しい」を十分満たしていると言える。

5. 活用チェック法の推敲支援への応用

前節までに述べた活用チェック法は、文章中にある用言の品詞と活用形を高速に推定するものである。活用チェック法の推敲支援への応用を考えると、1 つは前節でも述べたように、『推敲』に組み込まれている機能の抽出精度を向上させることである。もう 1 つは、従来の『推敲』にはない新しい推敲支援機能を開発することが考えられる。この節では、後者について述べる。

活用チェック法の推敲支援への応用として現在までに考えているものには、逆茂木型の文^[9]の検索、動詞の表記揺れの検出^[10]、並列構造の推定^[11]がある。これらについて以下で述べる。

5.1 逆茂木型の文の検索

日本語は修飾句・修飾節を修飾する語句の前側に置く、修飾句・修飾節前置型の構造の言語である。この修飾句・修飾節が長い文、または修飾節の中の言葉にさらに修飾節がかかるような文は読みづらくなる。文を木に例えると、文の概要が幹となり、修飾句・修飾節が枝となると考えられる。逆茂木型の文とは、枝がたくさん生い茂って、文の幹が分かりにくくなっている文を指す。修飾句・修飾節が複雑に絡まっているために、読み手に書き手の意図が通じにくくなっている。

逆茂木型の文の修飾節がどのような係り受け関係にあるかを正確に解析して書き手に指摘することは難しい。そのため、「連体修飾節が続く文」と「長い連体修飾節を含む文」とを逆茂木型の文として近似的に抽出する。これらは、共に連体形の用言の位置が分かれば抽出可能となるので、活用チェック法を文末から適応することで、連体修飾節を検出する。

5.2 動詞の表記揺れの検出

表記揺れの検出は推敲作業というより校正作業の部類に入るが、表記を揃えて見栄えをよくするために、表記揺れの検出も推敲作業の1つと言える。

動詞の表記揺れを検出するためには、まず文章中から動詞を抽出しなければならない。この動詞の抽出に活用チェック法を用いる。動詞を抽出した後に、表記揺れを検出するためには、読みの情報が必要になるので、「ふりが名表」と呼ぶものを用意した。これを用いることで、活用型が一致する同じ読みの動詞が存在する場合、それを表記揺れとして書き手に指摘する。

5.3 並列構造の指摘

並列構造とは、一文中に同等の機能を持つ複数個の文節列を並べた構造である。並列構造を含む文は、どの文節列とどの文節列が並列しているかという並列要素の認識において曖昧性を含みやすい。そのため並列構造を誤って認識してしまうことが多く、文の書き手と読み手の間に食い違いが生じやすい。

本研究では、並列構造のうち名詞並列と部分的並列^[12]を対象にして、構造的類似性に基づいて並列構造を推定する^[11]。しかし、並列構造を

表 4: 係り受け関係を決定する一般則と経験則

一般則	各文節はそれより文末側にある1つの文節に係り得る
	各文節はそれより文頭側にある0個以上の文節を受け得る
	係り受け関係は互いに交差しない
経験則	いくつかの例外を除いて、係り得る最も近い文節に係る
	読点を伴う文節は連体修飾文節には係り得ない

判断する単語が明示されている場合には、読み手はその単語をもとに並列要素を決定する。このことを反映するため、並列要素を決定する表層的な手掛かりおよび経験則を利用する。

名詞並列と部分的並列の並列構造を推定する手順は、前処理と本処理の2つに分けることができる。それぞれ以下の手順で処理を行う。

・前処理:

- (1) 主に字種情報を利用して、文を仮の文節に分割する。
- (2) 仮の文節に文節の性質を付与し、それを用いて仮の文節の分割誤りを修正する。

文節の性質とは、文節が持ち得る資格である。文節の性質は、係りの資格と受けの資格から成る。本研究では、係りの資格は各格要素、連体修飾要素、連用修飾要素、並列のキー要素など17種類に分類した。また、受けの資格は品詞と活用形をもとにして10種類に分類した。これらの文節の性質を求める際には、活用チェック法で求まる用言の品詞・活用形を利用する。

- (3) 文節の性質を利用して文節間の係り受け関係を決定する。

係りの資格と受けの資格の係り受け可能の関係を用いて可能な係り先を全て求める。表4に示した一般則と経験則を用いて係り先を1つに絞る。この処理を文末の文節から文頭に向かって順に行う。

・本処理: 前処理で求めた文節の性質と文節間の係り受け関係を用いて並列構造の推定を行う。

- (1) 文章中の並列のキーを検索し、各並列のキーの前後の文節列に対して並列構造になりうる最長の文節列を求める。

表 5: 並列のキーの抽出結果

抽出した並列のキー	1,028	
正しい並列のキー	979	(95.2%)
キーでないものをキーと判定	49	(4.8%)
文中の並列のキー	1,006	
抽出できた並列のキー	979	(97.3%)
キーの見逃し	27	(2.7%)

表 6: 並列要素の推定結果

正しい並列要素の文節数	キーを		総計
	1つ持つ	複数持つ	
正しく推定	331	394	725
誤って推定	102	152	254
第二種の誤り			49
推定精度 (%)	76.4	72.2	70.5

- (2) 構造的類似性に基づいて部分的並列の並列要素を推定する。
- (3) (2) で並列要素を決定できなかったものに対して、名詞並列の並列要素を推定する。
- (4) (1) で求めた最長の文節列内に複数の並列のキーを含む並列構造の推定を行う。

本推定法を計算機上に実装し、JICSTの抄録文(299件, 54,858文字)に対して名詞並列と部分的並列の並列構造の自動推定を行った。推定結果の判別は人手で行った。並列のキーの抽出結果表5にを示す。その結果、第一種の誤りを27個、第二種の誤りを49個犯していた。抄録文に含まれる並列のキーが1,006個で、本手法で指摘した並列のキーが1,028個であったので、適合率は95.2%であり、再現率は97.3%であることになる。

抽出した並列のキー1,028個に対して並列要素の推定を行った。その結果を表6に示す。単一並列構造の推定精度は76.4%(正解:331個, 不正解:102個)であり、複合並列構造の推定精度は72.2%(正解:394個, 不正解:152個)であった。単一並列構造、複合並列構造のいずれの場合でも、正しい並列構造の前置要素と後置要素のいずれも文節数が1である並列構造の推定精度が高いことがわかる。並列のキーの抽出において第二種の誤りを犯すので、本手法による推定精度を考える場合、第二種の誤りも考慮しなければならない。そのため、全体を総合

した抽出精度は70.5%(正解:725個, 不正解:254個, 第二種の誤り:49個)となる。

6. 日本語文章推敲支援ツール『推敲』

我々は、『推敲』を大型計算機上でまず開発した¹⁾。しかし、大型機には使用上の制約があったり、ユーザインタフェースが不十分であったりして、文章作成のような個人的な仕事を行うのには快適でない。そこで、機能強化やユーザインタフェースの向上を期して、『推敲』をパソコン(MS-DOS)上で実現した²⁾。パソコン版『推敲』は処理速度の速さと、ユーザインタフェース設計の自由度が大きいことも相まって、実用的なツールとなった。その後、ハードウェアの進歩と計算機の普及に伴って、ワークステーション(WS)やWindowsマシンが使われるようになって、WS版『推敲』やWindows版『推敲』の必要性が生じた。

『推敲』が果たすべき役割は、「文章中から問題となる箇所を指摘する」ことである。指摘された箇所をどう書き直すか、そのままにしておくかの判断は書き手に任せている。計算機と人間とが協調して推敲作業を行うことを想定している。そのために、書き手の思考を妨げない処理時間で処理できることを重要視して開発してきた。

『推敲』は、基本コマンドとマクロコマンドの2種類のコマンドインタフェースを持っている。基本コマンドは検索コマンド、集合演算コマンド、ソートコマンド、表示コマンド、その他のコマンドの5種類からなる。『推敲』は、これらの基本コマンドを組み合わせで推敲に役立つ情報を書き手に提供する。これらのコマンドはフォルダと呼ぶデータ構造を操作する。あるコマンドの結果をフォルダに格納し、そのフォルダの内容を入力として別のコマンドを実行する方法で基本コマンドを組み合わせる。

『推敲』の使用法に慣れたユーザであれば基本コマンドの組み合わせを工夫することで様々な情報を抽出することができるが、慣れないユーザはなかなかうまく行かない場合がある。そのため、基本コマンドを組み合わせで押し着せのコマンドとしてマクロコマンドを用意している。マクロコマンドの機能は、よく読まれている作文技術の本^{19, 8, 13)}を参考にして、計算機に肩代

表 7: マクロコマンドメニュー

1. 「推敲」終了	11. 受身
2. カタカナ表記のゆれ	12. 指示詞
3. アルファベット表記のゆれ	13. とりたて詞
4. 文末の特徴	14. 接続助詞「が」
5. 長すぎる文	15. 連用中止法
6. 長いひらがな列	16. 「は」を複数含む文
7. 文章の文字情報	17. 否定表現を複数含む文
8. 非対応括弧	18. 「ように」+否定表現
9. 使い方	19. 任意文字列検索
10. 基本コマンドモード	20. ユーザマクロメニュー

りしてもらえそうなものを揃えている。表 7 に WS 版「推敲」で用意しているマクロコマンドを示す。

パソコン版「推敲」を試用してもらった後の感想として、「編集機能があって欲しい」が非常に多かった。WS 版「推敲」を設計する際に、その点を検討するために解析部とユーザインタフェースとの分離を行った。これにより、ユーザインタフェースを mule などのエディタとすることによって、エディタとの連携を実現できる^[4]。解析部とユーザインタフェースとを別々のプロセスとして動作させ、プロセス間通信を用いて相互にデータのやりとりを行う。ユーザインタフェースを親プロセスとし、pty(疑似端末)を介して解析部を子プロセスとして起動する。この実現形態をとる場合に、ユーザインタフェースに要求される機能は以下の通りである。

- 子プロセスを起動でき、プロセス間で通信ができる。
- ユーザからの入力を受けられる。
- 複数のテキストを表示することができる。
- 複数の表示ウィンドウ間において、対応する部分を参照することができる。

現在の WS 版「推敲」では、ユーザインタフェースとして mule を用いている。

7. おわりに

日本語文章推敲支援ツール「推敲」は、書き手の思考を妨げない処理時間で処理できることを重要視して開発してきた。そのために、解析対象の文章を部分的に解析することで推敲に役立つ情報を抽出する方法の構築を行ってきた。接続助詞「が」の抽出において、活用チェック法

は「推敲」で使用していた従来の抽出法に比べて適合率を約 10% 向上させた。一方、再現率は 100% に保ったままであった。また、他の推敲支援項目の抽出にも有効であることが分かった。

参考文献

- [1] 牛島和夫, 日並順二, 尹志熙, 高木利久: “日本語文章推敲支援ツールのプロトタイプング”, コンピュータソフトウェア, Vol.3, No.1, 1986, pp.35-46.
- [2] 倉田昌典, 菅沼明, 牛島和夫: “日本語文章推敲支援ツール「推敲」のパソコン上での実用化”, コンピュータソフトウェア, Vol.6, No.4, 1989, pp.55-67.
- [3] 牛島和夫, 石田真美, 尹志熙, 高木利久: “日本語文章推敲支援ツールにおける受身形の抽出法”, 情報処理学会論文誌, Vol.28, No.8, 1987, pp.894-897.
- [4] 菅沼明, 石田朗子, 倉田昌典, 牛島和夫: “日本語文章推敲支援ツール「推敲」における字面解析手法とその評価”, 情報処理学会自然言語処理研究会, No.68, 1988, 68-8.
- [5] 菅沼明, 倉田昌典, 牛島和夫: “日本語文章推敲支援ツール「推敲」における否定表現の抽出法”, 情報処理学会論文誌, Vol.31, No.6, 1990, pp.792-800.
- [6] 菅沼明, 笠原晋, 牛島和夫: “字面解析による用言の活用形推定について”, 情報処理学会論文誌, Vol.37, No.6, 1996, pp.1007-1016.
- [7] 吉田将, 日高達, 稲永紘之, 田中武美, 吉村賢治: “公用データベース日本語単語辞書の使用について”, 九州大学大型計算機センター広報, Vol.16, No.4, 1983, pp.335-361.
- [8] 清水幾太郎: 論文の書き方, 岩波新書, 1959.
- [9] 木下是雄: “理科系の作文技術”, 中公新書, 1981.
- [10] 笠原晋, 菅沼明, 牛島和夫: “日本語文章推敲支援における動詞の表記揺れの検出”, 情報処理学会九州支部研究会, 1996, pp.242-250.
- [11] 山村広臣, 菅沼明, 牛島和夫: “日本語文における名詞句の並列構造の推定および推敲支援への適用”, 情報処理学会自然言語処理研究会, No.111, 1996, pp.7-14.
- [12] 首藤公昭, 吉村賢治, 津田健蔵: “日本語技術文における並列構造”, 情報処理学会論文誌, Vol.27, No.2, 1986, pp.183-190.
- [13] 本多勝一: “日本語の作文技術”, 朝日新聞社, 1976.
- [14] 西園誠, 菅沼明, 牛島和夫: “日本語文章推敲支援ツール「推敲」のワークステーション上での実現”, 情報処理学会自然言語処理研究会, 1993, pp.11-17.