形態素解析・読み付与プログラムの開発

目 次

第 1 章　概要 *1*

1.1　設計指針 *1*

1.2　機能概要 *2*

1.3　プログラム概要 *3*

第 2 章　プログラムの機能仕様 *4*

2.1　形態素解析機能 *4*

2.1.1　形態素区切り機能 *4*

2.1.2　品詞付与機能 *5*

2.2　読み付与機能 *5*

2.2.1　基本読み付与機能 *5*

2.2.2　数詞・助数詞読み付与機能 *5*

2.2.3　「々」読み付与機能 *5*

2.2.4　読み修正機能 *6*

第 3 章　プログラムの構造仕様 *7*

3.1　形態素解析サブプログラム *7*

3.1.1　形態素解析辞書の仕様 *8*

3.2　読み付与サブプログラム *8*

3.2.1　読み変化サブプログラム *9*

3.2.2　読み修正サブプログラム *14*

参考文献 *16*

第 4 章　納入物件一覧 *17*

第 5 章　プログラムの操作 *18*

5.1　インストール方法 *18*

5.1.1　形態素解析サブプログラム ChaSen のインストール *18*

5.1.2 読み変化サブプログラム ChaWan のインストール 18
5.1.3 読み修正サブプログラム PostProcess のインストール 19
5.2 操作手順 19
5.2.1 形態素解析サブプログラム ChaSen の操作 19
5.2.2 読み変化サブプログラム ChaWan の操作 19
5.2.3 読み修正サブプログラム PostProcess の操作 20

付 録 A 読み付与ガイドライン 21
A.1 読みの表記方法 21
A.2 区切りに関する基本方針 22
A.3 その他 23
A.4 読み付与プログラム出力例 25

付 録 B 日本語の形態素解析 28
B.1 形態素解析システム「ChaSen」 28
B.1.1 ChaSen の概要 28
B.1.2 ChaSen による形態素解析 28
B.1.3 ChaSen に関するその他の情報 32
B.2 ChaSen マニュアル 33
参考文献 77

付 録 C ChaWan のメンテナンス法 78

第1章　概要†

大語彙日本語連続音声認識のための統計的言語モデル作成においては、正しく形態素に区切られ、読みが付与された言語データが大量に必要となる。このため、大規模テキストコーパス等から得られる日本語の任意の文章を処理して、このような言語データを生成するプログラムの開発を行う。

1.1 設計指針

システムに科せられる要件は以下のとおりである。

(1) 任意の文章入力

(2) 形態素情報の正確性

(3) 読みの正確性

(1) および (2) については、既存の形態素解析器を利用することで実現を図る。ただし、(2) について、形態素の粒度には一般に一意な正解というものが存在せず、システムによって細かく区切ったり、大きなかたまりにしたりといった傾向がある。今回は大語彙日本語連続音声認識が最終的な目的であるため、これに適した形態素の粒度を想定し、形態素解析辞書や文法も必要に応じて作成することとする。

次に、(3) については、(2) で想定した形態素に対して読みを付与することにより実現する。このとき、正しい読みとは何かという問題が生ずる。ここでも、最終目的である大語彙日本語連続音声認識という観点から正解の基準を設定する。具体的には、実際の発声に近い読みの表記法として、NHK 日本語発音アクセント辞典 [1] に準拠したカタカナ表記を採用し、正解の基準も同辞典に準じる。ただし、鼻音、無声化、およびアクセントに関する情報は記述しない。これは、チ・ツの濁音をそれぞれジ・ズと表記したり、長音の表記に「ー」を採用しており、現代かなづかいに基づいた一般的な表記とは異なる。また、日本語の漢字かな混じり文では、同じ表記が複数の読みの可能性をもつことがある。今回の

† 山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

目的の下では、このような場合に、特定の読みを選択するのではなく、可能な読みはすべて付与されることが望ましい。このため、読みの併記方式を採り入れ、複数の読みが可能な場合には、それらを併記することとする。更に、連接する形態素にともに読みが併記される場合に、それらの間の組合せに一定の制約が課せられることがある。このような場合には、それらの形態素を結合して一つの形態素とし、あり得ない読みの組合せが生じないようにする。以上の読み表記法をまとめたものを付録 **5.2.3** に示す。

以上の考察から導かれるシステムの全体構成を図 **1.1** に示す。

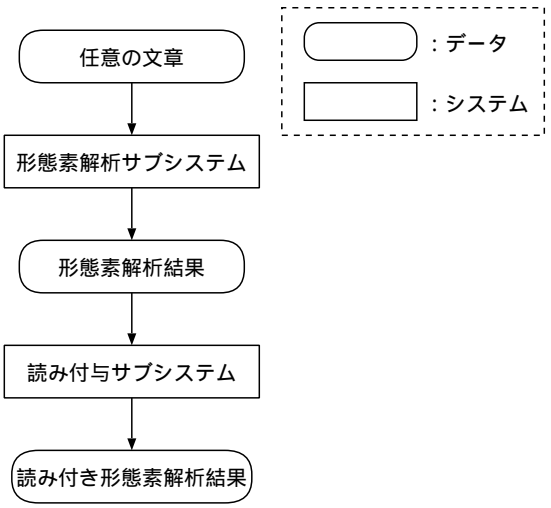

図 **1.1.** システムの全体構成

1.2 機能概要

本プログラムは、形態素解析機能と読み付与機能の二つの大機能を持ち、それぞれの大機能は以下に示す小機能を持つ。

(1) 形態素解析機能

(1-1) 形態素区切り機能

(1-2) 品詞付与機能

(2) 読み付与機能

(2-1) 基本読み付与機能

(2-2) 数詞・助数詞読み付与機能

(2-3) 「々」読み付与機能

(2-4) 読み修正機能

形態素解析機能は、形態素区切り機能と品詞付与機能から成り、入力文を解析し、形態素単位に区切って、それぞれの形態素に品詞を付与する。読み付与機能は、基本読み付与機能、数詞・助数詞読み付与機能、「々」読み付与機能、および読み修正機能から成り、各形態素に前後の接続から正しい読みを付与し、必要に応じて修正を行う。各機能の詳細については第 1.3 章で述べる。

1.3 プログラム概要

本プログラムは、形態素解析サブプログラムと読み付与サブプログラムから成り、このうち読み付与サブプログラムは、さらに読み変化サブプログラムと読み修正サブプログラムから成る。これらと前節で述べた各機能との対応を、以下に示す。

(1) 形態素解析サブプログラム
　　形態素区切り機能、品詞付与機能、基本読み付与機能

(2) 読み付与サブプログラム

　(2-1) 読み変化サブプログラム
　　数詞・助数詞読み付与機能、「々」読み付与機能

　(2-2) 読み修正サブプログラム
　　読み修正機能

各サブプログラムの詳細については第 2.2.4 章で述べる。

第2章　プログラムの機能仕様†

2.1 形態素解析機能

形態素解析機能は、入力文を解析し、形態素単位に区切って、それぞれの形態素に品詞を付与する。特に、当該形態素が活用語である場合は、活用型・活用形・基本型を特定する。このために、品詞体系、活用型・活用形体系、形態素辞書、接続表の四つの情報を用いる。このうち、品詞体系とは、文を構成する際の働きを基準として構築された日本語の語の分類である。実際には様々な体系が考えられているが、ここでは、情報処理振興事業協会 (IPA) で設定され、新情報処理開発機構 (RWCP) によるテキストデータベース報告書 [2] に掲載された IPA 品詞体系に基づき、その一部を修正して用いる。活用型・活用形体系とは、動詞・形容詞・形容動詞・助動詞などの活用語が、それぞれどのような活用型を持つか、また、各活用型がどのような活用形および活用語尾を持つかを記述したものである。形態素辞書とは、それぞれの品詞分類に属する各形態素について、その形態素の見出し語・品詞・活用型などを記述する。接続表とは、品詞もしくは単語の間の連接可能性を表形式で記述したものである。

本機能は形態素区切り機能、品詞付与機能の二つの小機能から成る。

2.1.1 形態素区切り機能

形態素区切り機能は、入力文を解析し、接続表に基づいた接続可能性の判定を行いながら、形態素辞書中に記述されている形態素の列として、入力文を形態素単位に区切る。このとき、形態素辞書中に存在しない形態素については、未定義語として取り扱えるような機能も有する。

† 宇津呂 武仁 (奈良先端科学技術大学院大学 情報科学研究科)、天白 成一 ((株) アルカディア)、山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

2.1.2 品詞付与機能

品詞付与機能は、入力文を構成する各形態素に品詞情報を付与する。この情報は品詞体系に基づき、形態素辞書中に記述されているものとする。また、当該形態素が活用語である場合に付与される活用型・活用形・基本型に関する情報は活用型・活用形体系により規定される。

2.2 読み付与機能

読み付与機能は、各形態素に前後の接続も考慮して正しい読みを付与する。付与される読みは付録 5.2.3 に示す読み付与ガイドラインに沿ったものである。

本機能は、基本読み付与機能、数詞・助数詞読み付与機能、「々」読み付与機能、および読み修正機能の四つの小機能から成る。

2.2.1 基本読み付与機能

基本読み付与機能は、入力文を構成する各形態素に対して、予め形態素毎に定められた読みを付与する。本機能により、同一形態素に対してはすべて同じ読みが与えられる。

2.2.2 数詞・助数詞読み付与機能

数詞・助数詞読み付与機能は、入力文中の数詞および助数詞に対して、規則に基づき、前後の接続から正しい読みを与える。ここでいう数詞とは、漢数字およびアラビア数字による表記を対象とし、読みにおける位取りの有無の判定機能を含む。また、数詞表記中の「 ,」「 .」「 −」「(」「)」「 ・」などの特殊記号の読みの取り扱いも対象とする。更に、数詞と助数詞が連接することによる一部の読みの変化にも対応する。

2.2.3 「々」読み付与機能

「々」読み付与機能は、入力文中の「々」「々々」に対して、前接する形態素の読みに応じて適切な読みを付与する。このとき、必要に応じて連濁現象にも対処する。

2.2.4 読み修正機能

読み修正機能は、現在の品詞体系、活用型・活用形体系では処理が不可能な形態素に対して、正しい読みを与える。このような例として、形容詞の連用ゴザイ接続の取り扱いがある。たとえば、「長うございます」という入力に対して、形容詞「長い」の語幹「長」の読みを「ナガ」として持っている限り、「ナゴ」という読みは得られない。本機能はこういった読みを修正するものである。

第3章　プログラムの構造仕様†

3.1 形態素解析サブプログラム

形態素解析サブプログラムは、任意の入力文を読み込み、形態素解析結果を出力する。本サブプログラムは、形態素区切り機能、品詞付与機能、基本読み付与機能を実現する。

本サブプログラムのエンジン部分には、既存の日本語形態素解析システムであるChaSen(茶筌)[3] を用いる。ChaSen は、文法の定義、単語間の連接関係の定義などを容易に変更できるように配慮して開発されたシステムであるため、今回の開発に最適であると判断した。ChaSen は奈良先端科学技術大学院大学情報科学研究科松本研究室がソフトウエア著作権を有するパブリックドメインソフトウェアである。

本サブプログラムの入力形式は、日本語2バイトコード (JIS またはEUC) で記述され、改行コードで一行一文に区切られたものとする。出力形式は、ChaSen の出力フォーマットを "%m\t%Y\t%M\t%h/%t/%f\n" または"%m\t%Y\t%M\t%P-\t%T*\t%F*\n" と設定した場合の出力とする。前者の場合は、品詞がコードで出力され、たとえば、

```
彼      カレ    彼      14/0/0
は      ワ      は      62/0/0
走っ    ハシッ  走る    44/17/8
た      タ      た      70/47/2
。      。      。      74/0/0
EOS
```

のような出力となる。後者の場合は、

```
彼      カレ    彼      名詞-代名詞-一般
は      ワ      は      助詞-係助詞
走っ    ハシッ  走る    動詞-自立       五段・ラ行      連用タ接続
た      タ      た      助動詞  特殊・タ        基本形
```

† 宇津呂 武仁 (奈良先端科学技術大学院大学 情報科学研究科)、天白 成一 ((株) アルカディア)、山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

```
。　　。　　。　　記号-句点
EOS
```

のような出力が得られる。

この前提の下に、ChaSen と同じく、奈良先端科学技術大学院大学で公開されている IPA 品詞体系に基づく ChaSen 用辞書セットのβ版 (ipadic1.0b2) をベースに、形態素解析辞書、活用体系、接続規則について独自の整備・拡張を行う。

3.1.1 形態素解析辞書の仕様

ipadic1.0b2 に含まれる形態素解析辞書に対して、以下の作業を行う。

辞書への発音表記の追加

辞書中の読みフィールドには、実際には現代かな遣いに基づく表記が格納されている。このために、新たに発音フィールドを設け、読み付与ガイドラインに沿った読みを格納する。

辞書のエントリの追加と削除、マージ

β版辞書から不要と考えられるエントリを削除する。また、固有名詞を中心に新たにエントリを追加し、解析能力の向上を図る。さらに、同一品詞で異なる読みによって複数エントリにわかれているものについては、それらをマージし、読みを併記した上で単一のエントリとする。例えば、「貴い」に対して「タットイ」「トートイ」という 2 種類の読みが存在する場合等である。この結果、原型と品詞の組に対して常に唯一のエントリが対応することになる。

コストの調整

解析済み正解コーパスを用いて、辞書中に持つ各エントリのコスト学習を行う。

3.2 読み付与サブプログラム

読み付与サブプログラムは、形態素解析サブプログラムの出力を読み込み、各形態素に対して形態素解析サブプログラムによって付与された読みの付け換えを行う。本サブプログラムは、読み変化サブプログラムと読み修正サブプログラムとから成る。

3.2.1 読み変化サブプログラム

読み変化サブプログラムは、形態素解析サブプログラムの出力を読み込み、品詞が「名詞-数」「名詞-接尾-助数詞」である形態素の読み変化、および形態素「々」「々々」に対する読み付与を行う。本サブプログラムは、数詞・助数詞読み付与機能、「々」読み付与機能を実現する。

数詞読み変化の方法

日本語の数詞の読み変化は、大きくは 2 通りの方法がある。一つは位取りを行うもので、もう一つは位取りを行わないものである。読み変化サブプログラムでは、位取りを行うか行わないかは自動的に判別する。また、数詞と助数詞が連接することで、一部の読みが音便化したり助数詞の先頭の音節が半濁音や濁音に変化する場合がある。このような現象をカバーすることが目的となる。

数詞の表記方法と読みの関係

日本語の文章においては、数字の表記形態が多数存在することは周知のことである。これは、従来より縦書きで漢数字を用いて記述されていたものが、明治以来、横書きでアラビア数字を用いて記載されるに至った歴史的な経緯も遠因である。そこで、日本語の文章中にみられる数字の表記方法に関して、文字の種別で類別するならば、大きく次の 3 つに分類することができる。

- ・アラビア数字のみによる表記　例: 198000
- ・漢数字のみによる表記　例: 十九万八千
- ・アラビア数字と漢数字の混在による表記　例: 198 千

また、この他にローマ数字による表記が与えられる場合があるが、ローマ数字による表記に必要な文字コードが一部の計算機システムでは、外字コードとして扱われているため、読み変化サブプログラムでは、ローマ数字を用いた数字表記に対する読み付与の処理を行わない。

数詞のみで構成されときは、先頭の文字が「〇 (ゼロ)」以外の文字から始まる場合は、普通に位取りして読む。ただし 16 桁まで、すなわち千兆までを読む。 16 桁以上の場合は、位取りせずに読む。

入力パターン	読み
０９３３	ゼロキューサンサン
１３０００	イチマンサンゼン
一万三千	イチマンサンゼン
1万3千	イチマンサンゼン
1万３０００	イチマンサンゼン
１３千	イチマンサンゼン

数字表記における特殊記号とその読み方

数字表記におけるもう一つの大きな課題は、特殊記号の取扱いである。これは、数字を表現するにあたって、数字だけではなく、特殊記号と合わせて使用することにより、意味を付加するために用いられる。数字表記における特殊記号としては、以下のようなものが考えられる。

, こんま KANMA
. てん DOT
ー ばー BAR
― ばー BAR
(かっこ KAKKO
) かっこ KOKKA
・ なかてん DOT

(1) 数字表記の中に特殊記号「,(こんま)」が入る場合

数字表記中に「,(こんま)」が入る場合は、いずれの場合も読み上げには影響しない。これは、数字の表記における欧米式の書式であり、通常のように位取りをして読む。

入力パターン	読み
２４，０９３３	ニジューヨンマンキューヒャクサンジューサン
４２，１９５千円	ヨンセンニヒャクジューキューマンゴセンエン
１２３，４５，６５	ヒャクニジューサンマンヨンセンゴヒャクロクジューゴ

最後のパターンは欧米式の数字表記おいては、誤りであるが、読み上げにおいては無視する。

(2) 数字表記の中に特殊記号「.(てん)」が入る場合

数字表記中に「．(てん)」が一個だけの場合は、小数点の表現として、位取りして読み、小数点以下は、位取りせずに読む。ただし、位を表す漢数字と混在している場合には、位取りを全体に施してから読む。一方、数字表記中に「．(てん)」が二個以上の場合は、位取りせずに読む。

入力パターン	読み
２４．０９３３	ニジューヨンテンゼロキューサンサン
４２．１９５千	ヨンマンニセンヒャクキュージューゴ
１２３．４５．６５	イチニーサンヨンゴロクゴ

(3)　数字表記の中に特殊記号「・(なかてん)」が入る場合

いずれの場合も「・(なかてん)」の処理は、「．(てん)」と同様に扱う。

入力パターン	読み
２４・０９３３	ニジューヨンテンゼロキューサンサン
４２・１９５千	ヨンマンニセンヒャクキュージューゴ
１２３・４５・６５	イチニーサンヨンゴロクゴ

(4)　数字表記の中に特殊記号「ー(ばー)」が入る場合

数字表記中に「ー(ばー)」または「—(ばー)」が入る場合に、数字は原則として、「マイナス」と読む。表記が、数前置詞の郵便記号 (〒) から始まる場合か、地名からの後処理として、数字を読む場合に限って、「の」と読む。

入力パターン	読み
〒５６２—００１３	ユービンバンゴーゴーロクニーノゼロゼロイチサン

(5)　数字表記の中に特殊記号「( ) (かっこ)」が入る場合

数字表記中に「( ) (かっこ)」が入る場合には、二通り考えられる。一つは、数字を単に括弧で括ったもの、もう一つは、電話番号の表記である。この二つの判別は、「( ) (かっこ)」に後続して、数字が付くかどうかで判断する。

入力パターン	読み
（２４）０９３３	ニーヨンゼロキューサンサン
（２４）	カッコニジューヨンカッコ

数詞と助数詞が連接した場合の読み変化

数詞の読み方は、後続する助数詞によって変化する。以下の表は、日本語発音アクセント辞典 [1] をもとに、これを分類したものである。

分類	一	二	三	四	五	六	七	八	九	十
A1	イチ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハチ	キュー	ジュー
A2	イチ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハチ	キュー	ジュッ
A3	イッ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハチ	キュー	ジュッ
A4	イチ	ニ	サン	ヨン	ゴ	ロッ	ナナ	ハチ	キュー	ジュッ
A5	イッ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハッ	キュー	ジュッ
A6	イッ	ニ	サン	ヨン	ゴ	ロッ	ナナ	ハチ	キュー	ジュッ
A7	イッ	ニ	サン	ヨン	ゴ	ロッ	ナナ	ハッ	キュー	ジュッ
B1	イチ	ニ	サン	シ	ゴ	ロク	シチ	ハチ	ク	ジュー
B2	イチ	ニ	サン	ヨン	ゴ	ロク	シチ	ハチ	ク	ジュー
B3	イチ	ニ	サン	ヨン	ゴ	ロク	シチ	ハチ	キュー	ジュー
B4	イチ	ニ	サン	ヨ	ゴ	ロク	シチ	ハチ	ク	ジュー
B5	イチ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハチ	ク	ジュー
B6	イチ	ニ	サン	ヨン	ゴ	ロク	ナナ	ハチ	キュー	ジュー
C1	ヒト	フタ	ミ	ヨ	イツ	ム	ナナ	ヤ	ココノ	ト
C2	ヒト	フタ	ミ	ヨ	イツ	ム	ナナ	ヤ	キュー	ト
C3	ヒト	フタ	ミ	ヨ	イツ	ム	ナナ	ハチ	キュー	ト
C4	ヒト	フタ	ミ	ヨ	イツ	ム	ナナ	ハチ	キュー	ジュー
C5	ヒト	フタ	ミ	ヨ	ゴ	ム	ナナ	ハチ	キュー	ジュッ
C6	ヒト	フタ	サン	ヨン	ゴ	ム	ナナ	ハチ	キュー	ジュー

また、数詞と助数詞の組み合わせによっては、助数詞の先頭の読み方が変化する場合がある。この現象は、通常の単語においては、連濁と呼ばれるような変化であるが、数詞と助数詞の組み合わせでは、単純な連濁の他に、複数の読み変化をとるパターンが存在する。例えば、「本 (ホン) 」では、「一本 (イッポン) 」、「二本 (ニホン) 」、「三本 (サンボン) 」というように読みが変化する。これに対応するためには、助数詞の方のデータとして、連濁するかどうかの情報を蓄えておく必要があり、これらの情報は一元的に管理される必要がある。

入力パターン	読み
1 個	イッコ
１６階	ジューロッカイ
3 国	サンゴク
3 分	サンプン
１０分	ジュップン

数詞および助数詞における読み併記

読み付与ガイドラインに従って読みを併記して出力させる場合に、数詞と助数詞に関しての読み付与においては、助数詞との間で読みが複数考えられるものは、併記して出力させる。また、読みを併記させる関係で、数詞と助数詞を分離せずに出力させる必要がある。

[例] 一日　　{ツイタチ/イチニチ}
　　一　　{イチ/イッ}
　　食　　ショク

読み併記を行う場合、以下のルールを考える。

1. A 型変化において助数詞の先頭が p,t,k,s の場合、イチ/イッ、ハチ/ ハッになる傾向が高い。
2. A 型、B 型変化において全てナナ/シチになる。
3. A 型、B 型変化において A1 型の標準的な読み方から変化するものは、すべて A1 型を許容する。
4. 「ジュッ」の読みでは「ュ」が脱落して、「ジッ」となりえる。
5. C 型、E1 型変化では、2 桁以上の読みは、A 型で変化する。

この他に読みを併記する場合としては、小数点を含む場合がある。

[例] ８０・６　　ハチ{ジュッ/ジッ}テンロク
　　９０・６　　キュー{ジュッ/ジッ}テンロク

「々」「々々」に対する読み付与

「々」および「々々」は前接する形態素によって読みがかわるため、これが独立の形態素として出現した場合は、前接する形態素の読みに応じて読みを付与する必要がある。このとき、読みによっては連濁を施す必要のあるものがある。

[例] 神　　カミ
　　々　　ガミ

3.2.2 読み修正サブプログラム

読み修正サブプログラムは、形態素解析サブプログラムの出力を読み込み、現在の品詞体系、活用型・活用形体系では処理が不可能な形態素に対して、正しい読みを与える。本サブプログラムは、読み修正機能を実現する。

現在の品詞体系、活用型・活用形体系で処理が不可能なものとは活用語の語幹の読みが変化する場合である。このうち、カ行変格活用動詞「来る」のように、特殊な活用型が予め割り当てられているものは、その活用形体系の中に語幹の読み変化まで含まれているため、この限りではない。読み修正サブプログラムで対象とするのは以下の場合である。

動詞「言う」の読み

「言う」は五段・ワ行促音便の動詞として分類されている。このとき、その読みは「イワ、イー、ユウ、イエ、イオ、イッ」のように変化するため、語幹「言」に対して一意な読みを割り当てることができない。そこで、辞書ではこの語幹に対しては「イ」という読みを割り当て、「言う」の場合のみ、これを「ユウ」に修正する。

形容詞・アウオ段・連用ゴザイ接続の読み

形容詞・アウオ段「長い」に「ございます」を後接すると、「長うございます」となるが、その読みは「ナゴーゴザイマス」である。すなわち、「長い」の語幹部分「長」の読みが「ナガ」から「ナゴ」に変化している。これも現在の品詞体系、活用型・活用形体系では処理できないので、辞書では「ナガ」という読みを割り当てておき、連用ゴザイ接続の場合にこれを「ナゴー」に修正する。

動詞・五段・カ行イ音便・連用タ接続の読み

例えば「聞く」という五段カ行イ音便の動詞の連用タ接続は、「聞いた」となるが、その読みは「キータ」というように長音化する。このとき、「キイ」から「キー」への修正を行う。

助動詞「う」の活用語尾への移行および長音化

　活用語に助動詞の「う」が後続する場合、その読みは長音化される。ところが、ここに形態素の切れ目があると、長音のみの形態素が生じるため、認識時に不都合を引き起こす。このため、助動詞-不変化型の「う」を活用語の活用語尾に追加して、意志・推量形という活用形とし、「う」の部分の読みを長音化する。

参考文献

[1] NHK 放送文化研究所: NHK 日本語発音アクセント辞典 新版 (1998).

[2] データベースワークショップ テキストグループ: テキストデータベース報告書, 技術研究組合 新情報処理開発機構 (1995).

[3] 松本裕治, 北内啓, 山下達雄, 今一修, 今村友明 : 日本語形態素解析システム『茶筌』version 1.0 使用説明書, Information Science Technical Report NAIST-IS-TR97007, Nara Institute of Science and Technology (1997).

第4章　納入物件一覧†

形態素解析・読み付与プログラムの納入物件は以下のとおりである。

(1) 形態素解析サブプログラム

ChaSen2.02 用辞書一式

(2) 読み付与サブプログラム

(2-1) 読み変化サブプログラム

ChaWan 2.06

(2-2) 読み修正サブプログラム

PostProcess 1.22

これらのプログラムは、UNIX OS 上で稼働する。また、ChaSen、ChaWan のコンパイルには C コンパイラが必要である。PostProcess の実行には Perl 処理系が必要である。

† 山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

第5章　プログラムの操作†

5.1 インストール方法

5.1.1 形態素解析サブプログラム ChaSen のインストール

ChaSen システムが存在するディレクトリを$CHASEN とする。

1. 環境に応じて$CHASEN/Makefile の BINDIR，LIBDIR，CC，CFLAGS などの項目を書き換えた後 “make” を実行する。これによって各プログラムがコンパイルされる。
2. “make dic” を実行する。これによって読み付与用形態素解析辞書・インデックスファイルが作成される。現在の読み付与用形態素解析辞書により最終的に生成されるファイルは

   $CHASEN/dic/chadic.int　約 9.5MB
   $CHASEN/dic/chadic.pat　約 1.3MB
   $CHASEN/dic/chadic.ary　約 0.9MB

   となる。
3. “make install” を実行する。デフォルトでは ,ChaSen 本体は/usr/local/bin/chasen に、辞書を作成するためのプログラムおよび辞書は/usr/local/lib/chasen/ 以下にインストールされる。

5.1.2 読み変化サブプログラム ChaWan のインストール

ChaWan システムが存在するディレクトリを $CHAWAN とする。

1. 環境に応じて $CHAWAN/Makefile の BINDIR，CC，CFLAGS などの項目を書き換えた後 “make” を実行する。これによってプログラムがコンパイルされる。

† 宇津呂 武仁 (奈良先端科学技術大学院大学 情報科学研究科)、天白 成一 ((株) アルカディア)、山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

2. “make install” を実行する。デフォルトでは ,ChaWan 本体は/usr/local/bin/chawan にインストールされる。

5.1.3 読み修正サブプログラム PostProcess のインストール

1. postprocess.pl の先頭行を実際に Perl インタプリタが存在するパスに書き換える。
2. postprocess.pl を実行パスが通っている任意のディレクトリにコピーする。

5.2 操作手順

5.2.1 形態素解析サブプログラム ChaSen の操作

ChaSen の実行は、標準入力、または引数で指定されたファイルから一行ごとに文を読み込んで処理を行う。大量の文に対して読み付与を行う場合には、一行一文のファイルを準備し、このファイル名を ChaSen の引数に指定する。なお、入力ファイル作成時に用いる 2 バイトコードは JIS または EUC とする。また、ChaSen は解析結果を標準出力に EUC コードで出力する。出力フォーマットは chasenrc で指定する。読み付与プログラム用には "%m\t%a0\t%M\t%h/%t/%f\n" または"%m\t%a0\t%M\t%P-\t%T*\t%F*\n" のいずれかのフォーマットを用いる。前者は品詞がコードで出力されるのに対し、後者は品詞名が日本語で表示される。ただし、言語モデル作成用には前者のフォーマットを用いる。

また、形態素解析辞書の変更をした場合は、その都度、辞書のコンパイル (make dic) とインストールを行う。

5.2.2 読み変化サブプログラム ChaWan の操作

ChaWan の実行は、標準入力、または-i 引数で指定したファイルから ChaSen の解析結果を読み込んで処理を行う。従って ChaWan が受け付ける 2 バイトコードは EUC コードのみである。入力形式は chasenrc において出力フォーマットを"%m\t%a0\t%M\t%h/%t/%f\n" または"%m\t%a0\t%M\t%P-\t%T*\t%F*\n" と指定したものに限る。前者の入力の場合は-c スイッチを指定する。ChaWan は読み変化の結果を入力と同じフォーマットで、標準出力、または-o 引数で指定されたファイルに出力する。

ChaWan のメンテナンス法を付録 **B.2** に示す。

5.2.3 読み修正サブプログラム PostProcess の操作

PostProcess の実行は、標準入力から ChaSen の解析結果を読み込んで処理を行う。従って PostProcess が受け付ける 2 バイトコードは EUC コードのみである。

入力形式は chasenrc において出力フォーマットを"%m\t%a0\t%M\t%h/%t/%f\n" または "%m\t%a0\t%M\t%P-\t%T*\t%F*\n" と指定したものに限る。前者の入力の場合は-c スイッチを指定する。PostProcess は読み変化の結果を入力と同じフォーマットで、標準出力に出力する。

よって、あるファイルに格納された文に読みをふり、結果を別のファイルに格納する一般的な操作は以下のようになる。

> chasen（入力ファイル名）| chawan（-c）| postprocess.pl（-c）>（出力ファイル名）

付 録A　読み付与ガイドライン†

A.1 読みの表記方法

原則として「NHK 日本語発音アクセント辞典 新版」の発音表記に従う。また、本辞典に記載されている発音を正解とする。

1. 発音表記はカタカナを用いる。これは発音を示すものであり、現代かなづかいによる表記とは必ずしも一致しない。

   （例）　または　マタワ
   　　綴り　ツズリ
   　　いう　ユウ
   　　アルミニウム　アルミニューム

   上例にあるように、「チ」「ツ」の濁音は「ジ」「ズ」とし、「ヂ」「ヅ」は用いない。また、助詞の「は」「へ」「を」の読みはそれぞれ「ワ」「エ」「ヲ」とする。
2. 漢語、和語、外来語を問わず、長音化して読まれる場合は長音記号「ー」で表記する。

   （例）　東京　トーキョー

3. 長音化される場合もされない場合もある場合はカナで表記する。

   （例）　覆う　オオウ
   　　経験　ケイケン

   例えば、「経験」は改まった場合は「ケイケン」、自然な発音では「ケーケン」となる。原則としてエ段に続くイは長音記号ではなく「イ」と表記する。
4. ガ行鼻音、および母音の無声化に関する情報は表記しない。
5. 不読記号の読みは表記のままとする。

† 山田 篤 ((財) 京都高度技術研究所/通信総合研究所)

（例）　▽　▽

6. 複数の読みが可能なものについてはそれらの読みを併記する。併記開始を‘{’、終了を‘}’、併記した読み間の区切りを‘/’で表す。

（例）　私　{ワタシ/ワタクシ}
行く　{イク/ユク}
平壌　{ピョンヤン/ヘイジョー}

八・一　{ハチ/ハッ}テン{イチ/イッ}
％　パーセント

１９７９　センキューヒャクナナジュー{キュー/ク}
年　ネン

A.2 区切りに関する基本方針

IPA/RWC の品詞体系に基づき、基本的にはなるべく分割して扱うが、例外についてはかためて扱う。かためて扱うものに関しては、全て明記する。

1. 数詞関連
   (a) 月名は「一月」~「十二月」までと「１月」~「１２月」までを一形態素とする。
   （例）　一月　イチガツ　一月　名詞-副詞可能
   (b) 「一つ」~「九つ」および「１つ」~「９つ」は一形態素とする。
   （例）　一つ　ヒトツ　一つ　名詞-一般
   (c) 日付はすべて数詞+「日」に区切る。
   （例）　十　トー　十　名詞-数
   日　カ　日　名詞-接尾-助数詞
   (d) ただし、「一日」「１日」は例外とし、形態素解析後、一つにかためたうえで「イチニチ」または「ツイタチ」の読みを与える。
   （例）　一　イチ　一　名詞-数
   日　ニチ　日　名詞-接尾-助数詞
   ----->
   一日　ツイタチ　一日　名詞-一般

2. 活用語の未然形+「う」はIPA/RWC品詞体系では分かれるが、これを形態素解析後、一つにかため、益岡・田窪文法の品詞体系による品詞情報を付与する。

（例）　だろ　ダロ　だ　助動詞　特殊・ダ　未然形
う　ウ　う　助動詞　不変化型　基本形
----->
だろう　ダロー　だろう　助動詞　助動詞だろう型　基本形

図ろ　ハカロ　図る　動詞-自立　五段・ラ行　未然ウ接続
う　ウ　う　助動詞　不変化型　基本形
----->
図ろう　ハカロー　図る　動詞-自立　五段・ラ行　意志形

3. 二つ以上の形態素が連続して複数の読みをもつが、それらの組合せに一定の規則がある場合は、形態素解析後、それらをかためたうえで、可能な組合せの読みを与える。

（例）　8　ハチ　8　名詞-数
分　フン　分　名詞-接尾-助数詞
----->
8分　{ハチフン/ハップン}　8分　名詞-一般

金　キム　金　名詞-固有名詞-人名-姓
日成　イルソン　日成　名詞-固有名詞-人名-名
----->
金日成　{キムイルソン/キンニッセイ}　金日成　名詞-固有名詞-人名

A.3 その他

1. 数詞の取り扱いについては、読み付与プログラムでは、統一的に数詞+ 助数詞のかたまりとして扱い、数詞内の構造 (例えば位取り) で区切るということは行わないが、これは言語モデルの作成時に表記が統一される。

（例）　１９７９　センキューヒャクナナジュー{キュー/ク}　１９７９　名詞-数

年　ネン　年　名詞-接尾-助数詞
---or---
一九七九　センキューヒャクナナジュー{キュー/ク}　一九七九　名詞-数
年　ネン　年　名詞-接尾-助数詞
---or---
千九百七十九　センキューヒャクナナジュー{キュー/ク}　千九百七十九　名詞-数
年　ネン　年　名詞-接尾-助数詞
----->
千　セン　千　名詞-数
九百　キューヒャク　九百　名詞-数
七十　ナナジュー　七十　名詞-数
九　{キュー/ク}　九　名詞-数
年　ネン　年　名詞-接尾-助数詞

2. 外部仕様として、数詞+助数詞にわかれないものを列挙する。

(a) 一月、１月、二月、２月、三月、３月、四月、４月、五月、５月、六月、６月、七月、７月、八月、８月、九月、９月、十月、１０月、十一月、１１月、十二月、１２月、一つ、１つ、二つ、２つ、三つ、３つ、四つ、４つ、五つ、５つ、六つ、６つ、七つ、７つ、八つ、８つ、九つ、９つ、
(b) 一日、１日、
(c) 二十歳、２０歳、
(d) 八杯、８杯、八敗、８敗、八拍、８拍、八泊、８泊、八箱、８箱、八班、８班、八犯、８犯、八版、８版、八匹、８匹、八俵、８俵、八票、８票、八分、８分、八編、８編、八遍、８遍、八歩、８歩、八本、８本、八羽、８羽、十羽、１０羽

このうち、(a) は、形態素解析辞書に登録しておき、形態素解析の段階で一語にしてしまうもの、(b) は、形態素解析では分かれているが、その後、かためて、数詞処理にはかけずに、統計的読み付与を行うもの、(d) は、助数詞部分が清音、半濁音のいずれかによって、数詞部分の読みが変わるので、かためてしまうもの、である。

3. 国名の慣用読みで認められているものについては併記する。

（例）　米　{ベイ/アメリカ}

ただし、文脈によって、慣用読みがなされない場合には併記しない。

（例）　日　ニチ
　　米　ベイ

4. 略記については、元の読みも併記する。

（例）　経企庁　{ケイキチョー/ケイザイキカクチョー}

ただし、略記が十分に通用しているものについてはこの限りではない。

（例）　通産省　ツーサンショー

5. 読まれない丸括弧については、形態素解析の後処理で、原文より除く。

A.4 読み付与プログラム出力例

```
----------
日本　{ニホン/ニッポン}　日本　名詞-固有名詞-地域-国
は　ワ　は　助詞-係助詞
この　コノ　この　連体詞
使用　シヨー　使用　名詞-サ変接続
済み　ズミ　済み　名詞-接尾-一般
核　カク　核　名詞-一般
燃料　ネンリョー　燃料　名詞-一般
の　ノ　の　助詞-連体化
中　ナカ　中　名詞-非自立-副詞可能
から　カラ　から　助詞-格助詞-一般
、　、　、　記号-読点
燃料　ネンリョー　燃料　名詞-一般
として　トシテ　として　助詞-格助詞-連語
有効　ユーコー　有効　名詞-形容動詞語幹
に　ニ　に　助詞-副詞化
使える　ツカエル　使える　動詞-自立　一段　基本形
プルトニウム　プルトニューム　プルトニウム　名詞-一般
を　ヲ　を　助詞-格助詞-一般
取り出す　トリダス　取り出す　動詞-自立　五段・サ
```

```
行　基本形
核　カク　核　名詞-一般
燃料　ネンリョー　燃料　名詞-一般
サイクル　サイクル　サイクル　名詞-一般
構想　コーソー　構想　名詞-サ変接続
を　ヲ　を　助詞-格助詞-一般
掲げ　カカゲ　掲げる　動詞-自立　一段　連用形
て　テ　て　助詞-接続助詞
いる　イル　いる　動詞-非自立　一段　基本形
。　。　。　記号-句点
EOS
----------
警察庁　ケイサツチョー　警察庁　名詞-固有名詞-組織
の　ノ　の　助詞-連体化
まとめ　マトメ　まとめ　名詞-一般
に　ニ　に　助詞-格助詞-一般
よる　ヨル　よる　動詞-自立　五段・ラ行　基本形
と　ト　と　助詞-接続助詞
、　、　、　記号-読点
総　ソー　総　接頭詞-名詞接続
検挙　ケンキョ　検挙　名詞-一般
件数　ケンスー　件数　名詞-一般
は　ワ　は　助詞-係助詞
二万三千五百九十八　ニマンサンゼンゴヒャクキュージュー{ハチ/ハッ}　二万三千五百九十八　名詞-数
件　ケン　件　名詞-接尾-助数詞
で　デ　だ　助動詞　特殊・ダ　連用形
、　、　、　記号-読点
逮捕　タイホ　逮捕　名詞-サ変接続
者　シャ　者　名詞-接尾-一般
は　ワ　は　助詞-係助詞
四十四　ヨンジューヨ　四十四　名詞-数
人　ニン　人　名詞-接尾-助数詞
```

```
。　　。　　。　　記号-句点
EOS
----------
```

付 録B　日本語の形態素解析†

B.1 形態素解析システム「ChaSen」

本節では、形態素解析システム「ChaSen」の具体的な利用方法の概略について説明する。なお、詳細については、付録 **B.2**「日本語形態素解析システム ChaSen マニュアル」を参照されたい。

B.1.1 ChaSen の概要

日本語形態素解析システム ChaSen(茶筌)(付録 **B.2**) [1] は、使用者によって文法の定義、単語間の連接関係の定義などを容易に変更できるように配慮して開発されたシステムで、以下の特徴を持つ。

- 特定の文法体系に依存しないシステム部分と文法・辞書の部分が独立。
- システム部分 — 文法・辞書記述部と形態素解析プログラム部からなる。
- 文法体系、単語間の連接関係の入れ換えが可能。
- 現在配布されているパッケージには、一般利用者のための標準的文法として、システム標準文法が用意されている。現在配布されている形態素辞書の語数は約 23 万 7 千語である。

B.1.2 ChaSen による形態素解析

まず、ChaSen がインストールされているディレクトリを`$CHASEN` とすると、ChaSen の実行ファイルは、`$CHASEN/chasen/chasen` である。

† 宇津呂 武仁 (奈良先端科学技術大学院大学 情報科学研究科)

ChaSen の実行

形態素解析の実行は、標準入力、または引数で指定されたファイルから一行ごとに文を読み込んで処理を行う。図 **B.1** の最初の例に示すように、適当な文を標準入力に与えると、形態素解析を行うことができる。

ChaSen 実行時のオプション

ChaSen 実行時のオプションとして、解が曖昧性を含む場合の表示方法に関する以下のオプションを試した結果を、図 **B.1** の二つ目以降の例に示す。

- `-b`　後方最長一致の解を一つだけ表示する (デフォルト)
- `-m`　曖昧性のある部分だけ、複数の形態素を表示する
- `-p`　曖昧性の組合せを展開し、すべての解を個別に表示する

オプション定義ファイル chasenrc の編集

1. chasenrc ファイルのコピー
   まず、`$CHASEN/chasenrc` ファイルを、各自のホームディレクトリの`.chasenrc` としてコピーする (ただし、各自のホームディレクトリが与えられていない場合は、コピー先を、ユーザごとに異なるファイル `chasenrc.xxx` とする。以下では、この場合を想定して説明を行う。)。
2. コスト幅の変更
   各自の `chasenrc.xxx` ファイルの許容コスト幅の指定を 4000 に変更して、形態素解析を行う例を図 **B.2** に示す。

   ```
   (コスト幅 4000)
   ```

3. 連結品詞の定義
   ある品詞の形態素が連続して出現したときに、一つの形態素として連結して出力させることができる。例えば、各自の `chasenrc.xxx` ファイルで

   ```
   (連結品詞 ((名詞 数)))
   ```

   と定義してあれば、連続した「数」が一つの形態素として出力されるようになる。連結品詞を定義した場合としない場合の形態素解析の例を図 **B.3** に示す。

```
% chasen
私は昨日学校へ行った。
私　{ワタ/ワタク}シ　私　名詞–代名詞–一般
は　ワ　は　助詞–係助詞
昨日　キノー　昨日　名詞–副詞可能
学校　ガッコー　学校　名詞–一般
へ　エ　へ　助詞–格助詞–一般
行っ　イッ　行く　動詞–自立　五段・カ行促音便　連用タ
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOS
% chasen -m
私は昨日学校へ行った。
私　{ワタ/ワタク}シ　私　名詞–代名詞–一般
は　ワ　は　助詞–係助詞
昨日　キノー　昨日　名詞–副詞可能
学校　ガッコー　学校　名詞–一般
へ　エ　へ　助詞–格助詞–一般
行っ　イッ　行く　動詞–自立　五段・カ行促音便　連用タ
行っ　オコナッ　行う　動詞–自立　五段・ワ行促音便　連用タ
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOS
% chasen -p
私は昨日学校へ行った。
私　{ワタ/ワタク}シ　私　名詞–代名詞–一般
は　ワ　は　助詞–係助詞
昨日　キノー　昨日　名詞–副詞可能
学校　ガッコー　学校　名詞–一般
へ　エ　へ　助詞–格助詞–一般
行っ　イッ　行く　動詞–自立　五段・カ行促音便　連用タ
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOP
私　{ワタ/ワタク}シ　私　名詞–代名詞–一般
は　ワ　は　助詞–係助詞
昨日　キノー　昨日　名詞–副詞可能
学校　ガッコー　学校　名詞–一般
へ　エ　へ　助詞–格助詞–一般
行っ　オコナッ　行う　動詞–自立　五段・ワ行促音便　連用タ
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOP
EOS
```

図 B.1. ChaSen による形態素解析の例

```
% chasen -m -r chasenrc.xxx
私は昨日学校へ行った。
私　{ワタ/ワタク}シ　私　名詞–代名詞–一般
は　ワ　は　助詞–係助詞
昨　サク　昨　接頭詞–名詞接続
昨日　キノー　昨日　名詞–副詞可能
日　ニチ　日　名詞–接尾–助数詞
日　{ニチ/ニッ}　日　名詞–固有名詞–地域–国
学　マナブ　学　名詞–固有名詞–人名–名
学　ガク　学　名詞–接尾–一般
学校　ガッコー　学校　名詞–一般
校　コー　校　名詞–接尾–一般
へ　エ　へ　助詞–格助詞–一般
行っ　イッ　行く　動詞–自立　五段・カ行促音便　連用タ
行っ　オコナッ　行う　動詞–自立　五段・ワ行促音便　連用タ
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOS
```

図 **B.2.** コスト幅 4000 の場合の形態素解析の例

(連結品詞 ((名詞 数))) を';' でコメントアウトした場合

```
% chasen -r chasenrc.xxx
五百万人いた。
五　ゴ　五　名詞–数
百　ヒャク　百　名詞–数
万　マン　万　名詞–数
人　ニン　人　名詞–接尾–助数詞
い　イ　いる　動詞–自立　一段　連用形
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOS
```

(連結品詞 ((名詞 数詞))) の場合

```
% chasen -r chasenrc.xxx
五百万人いた。
五百万　ゴヒャクマン　五百万　名詞–数
人　ニン　人　名詞–接尾–助数詞
い　イ　いる　動詞–自立　一段　連用形
た　タ　た　助動詞　特殊・タ　基本形
。　。　。　記号–句点
EOS
```

図 **B.3.** 連結品詞の定義をする/しない場合の形態素解析の例

B.1.3 ChaSen に関するその他の情報

ChaSen ホームページ

奈良先端科学技術大学院大学　松本研究室では、茶筌ホームページ (http://cl.aist-nara.ac.jp/lab/nlt/chasen.html) を公開しているので、最新の情報についてはこちらを参照されたい。

B.2 ChaSenマニュアル

NAIST Technical Report
NAIST-IS-TR99012

日本語形態素解析システム
『茶筌』version 2.0
使用説明書 第二版

松本裕治　北内啓　山下達雄　平野善隆　松田寛　浅原正幸

平成 11 年 12 月

Copyright © 1999 奈良先端科学技術大学院大学 松本研究室

Japanese Morphological Analysis System ChaSen 2.0 Users Manual
Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano and Hiroshi Matsuda
Copyright © 1999 Nara Institute of Science and Technology. All Rights Reserved.

Use, reproduction, and distribution of this software is permitted. Any copy of this software, whether in its original form or modified, must include both the above copyright notice and the following paragraphs.

Nara Institute of Science and Technology (NAIST), the copyright holders, disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness, in no event shall NAIST be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortuous action, arising out of or in connection with the use or performance of this software.

The Japanese morphological dictionary included in this system originates from ICOT Free Software. The following conditions for ICOT Free Software applies to the morphological dictionary of the system.

Each User may also freely distribute the Program, whether in its original form or modified, to any third party or parties, PROVIDED that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear on, or be attached to, the Program, which is distributed substantially in the same form as set out herein and that such intended distribution, if actually made, will neither violate or otherwise contravene any of the laws and regulations of the countries having jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to have agreed and understood, that there is no warranty whatsoever for the program and, accordingly, the entire risk arising from or otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other organization that participated in or was otherwise related to the development of the program and their respective officials, directors, officers and other employees shall be held liable for any and all damages, including, without limitation, general, special, incidental and consequential damages, arising out of or otherwise in connection with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

JUMAN	
version 0.6	17 February 1992
version 0.8	14 April 1992
version 1.0	25 February 1993
version 2.0	11 July 1994
ChaSen	
version 1.0	19 February 1997
version 1.5	7 July 1997
version 1.51	29 July 1997
version 2.0	15 December 1999
ChaSen for Windows	
version 1.0	29 March 1997
version 2.0	15 December 1999
NAIST Technical Report (NAIST-IS-TR99008)	
1st edition	20 April 1999
2nd edition	15 December 1999

目 次

1 茶筌の使用法 . . . 37
1.1 インストール手順 . . . 37
1.2 実行方法 . . . 38
1.3 実行時のオプション . . . 38
1.4 茶筌サーバとクライアントの使用法 . . . 39
1.5 出力フォーマット . . . 40
1.6 コマンドインタプリタ . . . 43
1.7 辞書の追加方法 . . . 44
2 chasenrc ファイル . . . 44
3 茶筌ライブラリ . . . 47
4 他のシステムからの利用 . . . 48
4.1 Prolog からの使用 . . . 48
4.2 Perl からの使用 . . . 50
4.3 Emacs からの使用 . . . 50
参考文献 . . . 51
付録 . . . 52
1 著作権および使用条件について . . . 52
2 JUMAN 2.0 から 茶筌 2.0 への拡張点 . . . 52
2.1 bi-gram 版と v-gram 版の相違点 . . . 52
2.2 茶筌 1.5 から 茶筌 2.0 への拡張点 . . . 53
2.3 茶筌 1.0 から 茶筌 1.5 への拡張点 . . . 53
2.4 JUMAN 2.0 から 茶筌 1.0 への拡張点 . . . 54
3 JUMAN3.0 と 茶筌 との関係について . . . 54
4 添付の日本語辞書 (ipadic2.0) の品詞体系について . . . 55
4.1 名詞 . . . 56
4.2 接頭詞 . . . 61
4.3 動詞 . . . 62
4.4 形容詞 . . . 68
4.5 副詞 . . . 71
4.6 連体詞 . . . 71
4.7 接続詞 . . . 71
4.8 助詞 . . . 71
4.9 助動詞 . . . 73
4.10 感動詞 . . . 75
4.11 記号 . . . 75
4.12 フィラー . . . 76
4.13 その他 . . . 76

はじめに

計算機による日本語の解析において，欧米の言語の解析と比べてまず問題になるのに次の 2 点があります．一つは形態素解析の問題です．ワードプロセッサの普及などによって日本語の入力には大きな問題がなくなりましたが，計算機による日本語解析では，まず入力文内の個々の形態素を認識する必要があります．これには実用に耐えられるだけの大きな辞書も必要であり，これを如何に整備するかという問題も同時に存在します．もう一つの問題として，日本語には広く認められ同意を得られた文法，ないし，文法用語がないという現実です．学校文法の単語分類および文法用語は一般には広く知られていますが，研究者の間ではあまり評判がよくありませんし，計算機向きではありません．

日本語の解析に真っ先に必要な形態素解析システムは，多くの研究グループによって既に開発され技術的な問題が洗い出されているにも係わらず，共通のツールとして世の中に流布しているものはありません．計算機可読な日本語辞書についても同様です．

本システムは，計算機による日本語の解析の研究を目指す多くの研究者に共通に使える形態素解析ツールを提供するために開発されました．その際，上の二つ目の問題を考慮し，使用者によって文法の定義，単語間の接続関係の定義などを容易に変更できるように配慮しました．

大学で小人数で開発したシステムであり，色々な点で不完全な部分があると思います．可能な限り順次改良を重ねる予定です．皆様の寛容な利用をお願いいたします．

本茶筌システムの原形は，京都大学長尾研究室および奈良先端科学技術大学院大学松本研究室において開発された日本語形態素解析システム JUMAN(version2.0) です．JUMAN は，京都大学および奈良先端科学技術大学院大学のスタッフおよび多くの学生の協力を得て作成したものです．また，辞書に関しては，Wnn かな漢字変換システムの辞書，および，ICOT から公開された日本語辞書を利用し，独自に修正を加えました．JUMAN 2.0 をともに開発した京都大学の黒橋禎夫氏，現在キャノン勤務の妙木裕氏には特に感謝いたします．

JUMAN 開発のきっかけを作って下さった京都大学長尾真先生に感謝します．JUMAN 開発に関して様々な形で協力していただいた奈良先端大宇津呂武仁氏に感謝します．奈良先端大の知念賢一氏には，茶筌システムの開発に関して多くの助言をいただきました．奈良先端大在学時の今一修氏，今村友明氏には茶筌 1.0 および茶筌 2.0 β版の開発の際に種々の助力をいただきました．両氏および茶筌の開発に協力いただいた松本研究室のメンバーに深く感謝します．奈良先端大の鹿野清宏教授を代表とする「日本語ディクテーション基本ソフトウェアの開発」グループの方々には，IPA 品詞体系辞書の大幅な整備を行っていただきました．特に，御尽力いただいた電子技術総合研究所の伊藤克亘氏，ASTEM の山田篤氏に感謝いたします．話し言葉の解析を中心にして辞書の整備に様々な助言をいただいた奈良先端大の伝康晴氏に感謝します．また，一人一人の名を挙げることはできませんが，JUMAN システムおよび茶筌システムに対して多くのコメントと質問をいただいた利用者の方々に感謝します．

平成 11 年 12 月 15 日

本システムに関するお問い合わせは以下にお願いします．
〒 630-0101
奈良県生駒市高山町 8916-5
奈良先端科学技術大学院大学
情報科学研究科 松本研究室
茶筌管理開発担当者集団
Tel: (0743)72-5240, Fax: (0743)72-5249
E-mail: chasen@cl.aist-nara.ac.jp

また以下の URL にて最新情報を提供しています．
URL: http://cl.aist-nara.ac.jp/lab/nlt/chasen.html

1 茶筌の使用法

1.1 インストール手順

1. 環境に応じて Makefile の BINDIR, LIBDIR, CC, CFLAGS などの項目を書き換えた後 'make' を実行する．これによって各プログラムがコンパイルされる．システム辞書を作成するためのプログラムは mkchadic/ 以下に，茶筌本体の実行ファイルは chasen/chasen に作成される．
   デフォルトでは v-gram 版がコンパイルされる．v-gram 版は従来の bi-gram 版とは辞書の形式などが異なる．bi-gram 版をコンパイルするには 'make bigram' を実行する．なお，すでにインストールされている ChaSen 1.51 などの古いバージョンも残して使い分けるには，Makefile の CHASEN, LIBDIR を

```
CHASEN = chasen2
LIBDIR = /usr/local/lib/chasen2
```

   などのように書き換えた後 'make' を実行すればよい．
2. 'make dic' を実行する．これによってシステム辞書が作成される．システム辞書作成に要する時間は，辞書の大きさとマシンの性能にもよるが，現在の辞書では，SPARCstation20 上で 2 ~ 3 分である．現在の標準辞書により最終的に生成されるファイルとその容量は以下の通り．

dic/chadic.int	約 9.5MB
dic/chadic.pat	約 1.3MB
dic/chadic.ary	約 0.9MB

3. 'make install' を実行する．このとき，

```
make[1]: [install] Error 1 (ignored)
```

   というようなメッセージが出力されるかもしれないが無視してよい．各ファイルは，デフォルトでは以下の場所にインストールされる．

/usr/local/bin/chasen	茶筌の実行ファイル
/usr/local/lib/chasen/chasenrc	chasenrc ファイル
/usr/local/lib/chasen/mkchadic/	システム辞書を作成するためのプログラム群
/usr/local/lib/chasen/dic/	文法・辞書ファイル

   ただし，以下のものはインストールされない．

chasen/libchasen.a	茶筌ライブラリ
chasen/chasen.el	Emacs 用インタフェース
prolog/*.pl	Prolog 用インタフェース
perl/ChaSen.pm	Perl モジュール

   なお，すでに ChaSen 1.51 などの古いバージョンがインストールされている場合は，

```
# rm -rf /usr/local/lib/chasen
```

   を実行してファイルを削除してから 'make install' を実行するとよい．
4. ユーザ専用の chasenrc ファイルを利用するには，dic/chasenrc をユーザのホームディレクトリに '.chasenrc' という名前でコピーする．なお，ChaSen 1.51 と使い分けたいときは '.chasen2rc' という名前でコピーすればよい．
   .chasenrc 内の「文法ファイル」に文法辞書が存在するディレクトリ，「PATDIC」に辞書ファイルから拡張子を取り除いたものを指定する．標準では以下のように設定すればよい．

```
(文法ファイル /usr/local/lib/chasen/dic)
(PATDIC chadic)
```

1.2 実行方法

システムの実行ファイルは，‘make’ によって chasen/chasen に作成される．また，‘make install’ によって/usr/local/bin/chasen などにインストールされる．

- 形態素解析の実行

  茶筌は，以下のように chasen コマンドを実行することにより起動される．

  ```
  % chasen [オプション] [ファイル名...]
  ```

  標準入力，または引数で指定されたファイルから一行ごとに文を読み込んで形態素解析処理を行なう．文字コードとしては日本語 EUC あるいは JIS(ISO-2022-JP) を受け付ける．

- 処理内容

  コスト最小 (それぞれの形態素の区切りで最小コストとの差が許容されるコスト幅以内) の解を求め，結果をオプションに従って表示する．出力時の文字コードは日本語 EUC である．各オプションの意味は次節にまとめる．

- 使用例

  入力ファイルを引数として指定できる．以下に使用例を示す．

```
% cat temp
私は昨日学校へ行きました．
% chasen temp
私      ワタクシ    私      名詞-代名詞-一般
は      ハ          は      助詞-係助詞
昨日    キノウ      昨日    名詞-副詞可能
学校    ガッコウ    学校    名詞-一般
へ      へ          へ      助詞-格助詞-一般
行き    イキ        行く    動詞-自立          五段・カ行促音便    連用形
まし    マシ        ます    助動詞             特殊・マス          連用形
た      タ          た      助動詞             特殊・タ            基本形
．      ．          ．      記号-句点
EOS
```

1.3 実行時のオプション

形態素解析の実行については，いくつかのオプションが用意されている．以下にそれをまとめる．-r など引数をともなうオプションでは，オプションと引数の間には空白があってもなくてもかまわない．

- 茶筌の起動についてのオプション

-s	茶筌サーバの起動
-P port	茶筌サーバのポート番号の指定 (-s オプションとともに使用，デフォルト値は 31000)
-D host[:port]	茶筌サーバに接続
-R	chasenrc ファイルを読み込まない (-D オプションとともに使用)
-a	環境変数 CHASENSERVER があってもスタンドアロンで実行

- 解が曖昧性を含む場合の表示方法 (曖昧性がない場合はどの方法も同じ表示となる)

-b	後方最長一致の解を一つだけ表示する (デフォルト)
-m	曖昧性のある部分だけ，複数の形態素を表示する
-p	曖昧性の組合せを展開し，すべての解を個別に表示する

- 各形態素の表示方法

-f	カラムを整えて表示 (デフォルト)
-e	完全な形態素情報を文字で表示
-c	完全な形態素情報をコードで表示
-d	各形態素を Prolog の複合項で表現し，それらをリストにしたものを出力
-v	美茶のための詳細表示
-F format	形態素を format で指定された形式で出力
-Fh	-F オプションの出力フォーマットのヘルプを表示

- その他

-j	句点あるいは空行を文の区切りとして解析
-o file	解析結果出力ファイルを指定
-w width	コスト幅を指定
-C	コマンドインタプリタを使用
-r rc_file	rc_file を chasenrc ファイルとして使用
-L lang	言語を指定
-lp	品詞番号と品詞名のリストを表示
-lt	活用型番号と活用型名のリストを表示
-lf	活用型番号，活用形番号と活用形名のリストを表示
-h	ヘルプメッセージを出力
-V	茶筌のバージョンを出力

-j オプションについて

茶筌では通常，改行をもって一つの入力文字列の終了とする．そのため，文の途中で改行が挿入されているファイルを解析した場合，正しい結果が得られなくなることが多い．

そのようなときは -j オプションをつけると，句読点など (デフォルトでは「。!? 」の 4 文字) あるいは空行を文の区切りとして解析を行うようになる．

また，茶筌 1.5 以降では chasenrc ファイルの「区切り文字」の項目を指定することにより，-j オプションをつけた時の文の区切り文字を設定することができる．

1.4 茶筌サーバとクライアントの使用法

茶筌 1.5 以降では，クライアントから茶筌サーバに接続して形態素解析を行うことができる．茶筌サーバを利用するには，まず，以下のように入力してサーバを起動する．

```
% chasen -s
```

次に -D オプションをつけて茶筌を実行すると，クライアントが起動しサーバに接続して形態素解析を行う．ここで，server は茶筌サーバを立ち上げたマシンのホスト名である．

```
% chasen -Dserver filename
```

茶筌サーバでは，デフォルトでポート番号 31000 を使う．このポート番号を変更するには，-P オプションをつけて茶筌サーバを起動する．

```
% chasen -s -P31234
```

また，クライアントの起動時には -D オプションでサーバ名の後ろに ':'(コロン) をつけ，ポート番号を指定すればよい．

```
% chasen -Dserver:31234 filename
```

また，以下のように環境変数 CHASENSERVER を設定することにより，-D オプションをつけなくても茶筌のクライアントが起動されるようになる．

```
% setenv CHASENSERVER server:31234
```

逆に，環境変数 CHASENSERVER を設定している場合でもスタンドアロンで茶筌を実行したいときは，-a オプションをつけて起動すればよい．

茶筌サーバを起動すると，スタンドアロンの時と同じように chasenrc ファイルを読み込み，辞書や文法ファイルが読み込まれる．クライアント側ではサーバが読み込んだ辞書と文法ファイルが用いられる．つまり，クライアントの起動時には chasenrc ファイルの「文法ファイル」「PATDIC」の項目は無視され，それ以外の項目だけが有効となる．もし，辞書や文法ファイルだけでなく他の項目についてもサーバと同じ設定でよければ，クライアントの起動時に -R オプションをつけることで chasenrc ファイルを読み込まずに解析を行わせることができる．

1.5 出力フォーマット

-F オプションや，chasenrc ファイルの「出力フォーマット」で出力フォーマットを指定することにより，解析結果の出力形式を変えることができる．

出力フォーマットの文字列の末尾に '\n' があれば，各形態素情報の表示ごとに改行を行い，文末の次に 'EOS' の 1 行を出力する．末尾に '\n' がなければ，1 文中の形態素情報を 1 行で出力し，行末に改行を表示する．

また，出力フォーマットに '-f', '-e', '-c' を指定すると，それぞれ -f, -e, -c と同じ出力形式になる．

出力フォーマットの使用例をいくつかあげる．

- デフォルト (-f オプション) と同様の出力 (v-gram 版の場合)

  "%m\t%y\t%M\t%U(%P-)\t%T␣\t%F␣\n" または "-f"

- 見出し，読み，品詞をタブで区切って表示 (v-gram 版の場合)

  "%m\t%y\t%P-\n"

- 見出し語のみ

  "%m\n"

- 分かち書き (見出し語を空白で区切って表示)

  "%m␣"

- 漢字かな変換

  "%y"

- ルビつき表示．“漢字 (かな)” の形式で表示する．

  "%r␣()"

出力フォーマットの変換文字の一覧を以下に示す．

変換文字	機能
%m	見出し (出現形)
%M	見出し (基本形)
%y, %y1	読みの第一候補 (出現形)(※ 1)
%Y, %Y1	読みの第一候補 (基本形)(※ 1)
%y0	読み全体 (出現形)
%Y0	読み全体 (基本形)
%a	発音の第一候補 (出現形)
%A	発音の第一候補 (基本形)
%a0	発音全体 (出現形)
%A0	発音全体 (基本形)
%rABC	ルビつきの見出し (“A 漢字 B かな C” と表示)(※ 2)
%i	意味情報
%Ic	意味情報 (空文字列か “NIL” なら文字 c)(※ 2)
%Pc	各階層の品詞を文字 c で区切った文字列 (v-gram 版のみ)
%Pnc	1 ~ n(n:1 ~ 9) 階層目までの品詞を文字 c で区切った文字列 (v-gram 版のみ)
%h	品詞の番号
%H	品詞文字列
%Hn	n(n:1 ~ 9) 階層目の品詞 (なければ最も深い階層)(v-gram 版のみ)
%b	品詞細分類の番号 (v-gram 版の場合は 0)
%BB	品詞細分類 (なければ品詞)
%Bc	品詞細分類 (なければ文字 c)(※ 2)
%t	活用型の番号
%Tc	活用型 (なければ文字 c)(※ 2)
%f	活用形の番号
%Fc	活用形 (なければ文字 c)(※ 2)
%c	形態素のコスト
%S	解析文全体
%pb	最適パスであれば “*”, そうでなければ “␣”
%pi	パスの番号
%ps	パスの形態素の開始位置
%pe	パスの形態素の終了位置 +1
%pc	パスのコスト
%ppiC	前に接続するパスの番号を文字 C で区切り列挙
%ppcC	前に接続するパスのコストを文字 C で区切り列挙
%?B/STR1/STR2/	品詞細分類があれば STR1，なければ STR2(※ 3)
%?I/STR1/STR2/	意味情報が “NIL” でも “”(空文字列) でもなければ STR1，そうでなければ STR2(※ 3)
%?T/STR1/STR2/	活用があれば STR1，なければ STR2(※ 3)
%?F/STR1/STR2/	%?T/STR1/STR2/ と同じ
%?U/STR1/STR2/	未知語なら STR1\，そうでなければ STR2(※ 3)
%U/STR/	未知語なら”未知語”，そうでなければ STR(%?U/未知語/STR/と同じ)(※ 3)
%%	% そのもの

変換文字	機能
.	フィールド幅の指定
-	フィールド幅の指定
1-9	フィールド幅の指定
\n	改行文字
\t	タブ
\\	\ そのもの
\’	’ そのもの
\"	" そのもの

※ **1** 茶筌付属の ipadic では「行く (いく/ゆく)」のように形態素が複数の読みを持つ場合，その読みを「{ イ/ユ } ク」のように，半角のブレースとスラッシュを使って表している．通常の読みの出力 (出力フォーマットの %y) では，その第一候補である「イク」が出力され，%y0 を使うと読み全体である「{ イ/ユ } ク」が出力される．

※ **2** A,B,C,c が空白文字の時は何も表示しない

※ **3** ‘/’ には任意の文字が使える．また，括弧 “(){}[]<>” を用いることもできる．以下に例をあげる．

- %?T#STR1#STR2#
- %?B(STR1)(STR2)
- %?U{STR1}/STR2/
- %U[STR]

1.6 コマンドインタプリタ

-C オプションにより茶筌実行時に以下のコマンドを対話的に与えることができる．

コマンド	機能
#V	-V オプションと同じ．
#F␣[文字列]	-F オプションと同じ．ただし，フォーマット文字列をクオートする必要はない．
#w␣[数字]	コスト幅の変更．例： #w␣500
#i	様々な情報の出力．
#e␣[単語]	辞書に単語が入っているか調べる．例： #e␣茶筌
#q	茶筌の終了．
#a	辞書への単語の追加．
#f	単語を追加する辞書の指定．
#s	単語追加後のパトリシア木のセーブ．

注意

- #a, #f, #s は，茶筌の解析結果を視覚化するプログラム「美茶」用のコマンドなので，無闇に実行してはならない．
- コマンドと引数の間は必ずスペース一文字でなければならない．

1.7 辞書の追加方法

ここでは辞書に新しい単語を登録する方法を述べる．

単に単語を登録するだけなら，dic/ に拡張子が .dic のファイル名をもつ辞書ファイルを追加し，再度 'make', 'make install' を実行すればよい．

特定のユーザ専用の辞書を作成して使用したい場合は，ユーザ辞書専用のディレクトリを用意し，そこに Makefile をコピーして 'make' を実行する．以下に例をあげる．

```
% mkdir ~/mydic
% cd ~/mydic
% cp /usr/local/lib/chasen/dic/Makefile .
(Noun2.dic をエディタなどで編集)
% make
```

'make' を実行すると chadic.int, chadic.pat, chadic.ary が作成される．

次に，ホームディレクトリにある .chasenrc 中で，以下のように「PATDIC」にユーザ辞書を追加する．「文法ファイル」は変更しなくてよい．

```
(文法ファイル /usr/local/lib/chasen/dic)
(PATDIC chadic
        /home/rikyu/mydic/chadic)
```

2 chasenrc ファイル

chasenrc ファイルは形態素解析プログラムに必要な様々な選択肢を定義するために用いられる．これらの定義は通常，/usr/local/lib/chasen/dic/chasenrc に記述されるが，利用者のホームディレクトリにある '.chasenrc' というファイルに記述するもできる．起動時オプションなどによって chasenrc ファイルを指定することもできる．具体的には次のような優先順位で chasenrc ファイルが読み込まれる．

1. 起動時に -r オプションで指定されたファイル．
2. 環境変数 CHASENRC で指定されたファイル．
3. 利用者のホームディレクトリにある .chasenrc．
4. 茶筌インストール時にインストールされた chasenrc ファイル．通常は /usr/local/lib/chasen/chasenrc．

設定項目一覧を以下に示す．このうち「PATDIC」または「SUFDIC」，「未知語品詞」，「品詞コスト」は必ず指定しなければならない．

1. 文法ファイルのディレクトリ
   文法ファイル (grammar.cha, ctypes.cha, cforms.cha, connect.cha.c) が存在するディレクトリを指定する．

   ```
   (文法ファイル /usr/local/lib/chasen/dic)
   ```

   「文法ファイル」は省略することができ，その場合 chasenrc ファイルがあるディレクトリと同じディレクトリを指定したとみなされる．茶筌に付属の辞書 ipadic1.01 以降では「文法ファイル」は省略されている．
2. システム辞書
   システム辞書 (chadic.int) とインデックスファイル (chadic.pat または chadic.ary) を，ファイル名から末尾の拡張子を除いたものを記述することによって指定する．複数組みの辞書を指定することもでき

る．また，相対パス，つまり “/” で始まらないパスを記述すると，文法ファイルと同じディレクトリを指定したとみなされる．例えば以下のように指定する．

```
(PATDIC chadic
        /home/rikyu/mydic/chadic)
```

この例では，以下の二組の辞書を指定している．

(a) 文法ファイルと同じディレクトリにある chadic.int, chadic.pat

(b) /home/rikyu/mydic/ にある chadic.int, chadic.pat

辞書引きに際しては，これらの辞書の両方が用いられる[1]．

辞書引きに SUFARY [2]を使う場合は「SUFDIC」を指定する．

```
(SUFDIC /usr/local/lib/chasen/dic/chadic)
```

SUFDIC は PATDIC に比べ，最初にインデックスファイルを読み込む時間は短いが，検索速度自体は遅いという特徴がある．解析時間を短くするには，解析文の量が少ないときは SUFDIC，多いときは PATDIC を使うとよい．

使用する辞書の最大数は，PATDIC, SUFDIC とも 5 個に設定されている．これを変更したい場合は，chasen/pat.h の MAX_DIC_NUMBER の値を変更してコンパイルしなおせばよい．

3. 未知語の品詞

未知語が発見された時に，その語をどのような品詞として連接規則を適用するかを指示する．複数の品詞を指定した時は，それぞれの品詞について連接規則が適用される．

```
(未知語品詞 (名詞 サ変接続))                ; 1 個の品詞を指定
(未知語品詞 (名詞 サ変接続) (名詞 一般))    ; 複数の品詞を指定
```

4. 品詞のコスト

形態素解析プログラムでは，解析結果の優先情報をコストとして計算している．解析に曖昧性がある場合は，コストの総計が低いものを優先することになっている．「品詞コスト」では，各品詞のコストの倍率と，「未知語」についてのコストを定義する．コストは正の整数値をとる．

```
(品詞コスト
        ((*)                1)
        ((未知語)           500)
        ((名詞)             2)
        ((名詞 固有名詞)    3)
)
```

同じ品詞に対してコストの定義が複数回指定されている場合は，後のものが優先される．上の例では，「名詞」の形態素のコストは 2 倍になるが，「名詞-固有名詞」以下に細分類される名詞の形態素のコストは 3 倍になる．また，先頭の ‘(*)’ の指定により，ここで明示的に定義されていない形態素のコストはすべて 1 倍 (そのままのコスト値) となる．未知語の形態素のコスト値はすべて 500 になる．

5. 連接コストと形態素コストの相対的な重みの定義

形態素解析におけるコストの計算は形態素のコストと連接のコストの総計として計算される．これら二種類のコストに異なる重みを掛けたい場合には，それを指定することができる．解析結果のコストはそれぞ

[1] 一組の辞書には同一の形態素の登録は行なわれないが，複数の辞書に同じ形態素が登録されている場合はあり得る．このような場合は，同じ形態素が複数得られることになる．

[2] SUFARY は文字列検索パッケージである．詳しくは http://cl.aist-nara.ac.jp/lab/nlt/ss/ を参照．

れのコストにここで指定された重みを乗じた値の総計として計算される．省略した場合の重みは 1 である．

```
(連接コスト重み 1)      ; デフォルト値
(形態素コスト重み 1)    ; デフォルト値
```

また，形態素解析の過程において，常にコストが最低の結果を出すのではなく，ある程度のコスト幅を許容したい場合がある．この許容幅を指定することができる．コスト幅におさまるすべての解を出力するには -m オプションを使う．

```
(コスト幅 0)    ; デフォルト値
```

6. 未定義連接コストの定義
   連接規則ファイルに連接規則が定義されていない形態素間の連接コストを指定する．未定義連接コストを設定しないか，あるいは 0 を指定すると，連接規則が定義されていない形態素どうしは決して連接しないという意味になる．

```
(未定義連接コスト 500)
```

7. 出力フォーマット
   出力フォーマットを指定することにより，解析結果の出力形式を変えることができる．

```
(出力フォーマット "%m\t%y\t%P-\n")
```

   詳しくは 1.5 節を参照のこと．
8. **BOS** 文字列
   解析結果の文頭に表示する文字列を指定する．“%S” を使うと解析文全体を表示できる．デフォルトは空文字列 (つまり何も表示しない)．

```
(BOS 文字列 "解析文: [%S]\n")
```

9. **EOS** 文字列
   解析結果の文末に表示する文字列を指定する．“%S” を使うと解析文全体を表示できる．デフォルトは “EOS\n”．

```
(EOS 文字列 "文末\n")
```

10. 空白品詞
    茶筌は，半角の空白文字 (ASCII コード 32) とタブ (ASCII コード 9) を空白とみなし，これらを無視して解析する．通常は，解析結果に空白の情報を出力しないが，「空白品詞」を設定することにより，空白についての情報を出力するようになる．例えば，

```
(空白品詞 (記号 空白))
```

    のように設定すると，空白を「記号-空白」として出力する．
    なお，出力フォーマットを “%m” に設定して，空白品詞を指定する (品詞は何でもよい) と，解析文と全く同じ出力が得られることになる．
11. 注釈
    ある文字列で始まりある文字列で終わる文字列を注釈のように扱い，その文字列の部分を無視して解析させることができる．解析結果には，その文字列が一つの形態素として出力される．

chasenrc ファイルには，出力時の品詞名，開始文字列，終了文字列からなるリストを記述する．終了文字列は省略することができ，その場合，開始文字列と一致する文字列自身を注釈として扱う．例えば，

```
(注釈 (("<" ">") (特殊 記号))
      (("「") (特殊 記号))
      (("」") (特殊 記号))
      (("\"" "\"") (名詞 引用文字列))
      (("[ " " ]"))
      )
```

と指定すると，以下のように解析される．

- <img src="cha.gif"> のように “<” で始まり “>” で終わる文字列を「特殊-記号」として出力．
- “「” あるいは “」” を「特殊-記号」として出力．
- "hello(again)" のようにダブルクォーテーションで囲まれた文字列を「名詞-引用文字列」として出力．
- [ちゃせん] のように “[” で始まり “]” で終わる文字列を無視して解析し，解析結果にはその文字列の情報は表示しない．

12. 連結品詞

ある品詞の形態素が連続して出現したときに，一つの形態素として連結して出力させるときに使用する．例えば，

```
(連結品詞 ((複合名詞) (名詞) (接頭詞 名詞接続) (接頭詞 数接続))
  ((記号)))
```

と定義すると，以下のように品詞を連結する．

(a) 連続した「名詞」「接頭詞-名詞接続」「接頭詞-数接続」を連結し「複合名詞」として表示する．なお，「複合名詞」は品詞定義ファイル grammar.cha に記述しておく必要がある．

(b) 連続した「記号」を連結し，「記号」として表示する．

13. 区切り文字

-j オプションをつけた時の文の区切り文字を並べ，一つの文字列にしたものを指定する (1.3 節参照)．区切り文字には全角文字，半角文字の両方を使用することができる．例えば

```
(区切り文字␣"。、，!? .,!?␣")
```

と定義すると，全角文字の「。、，!? 」のいずれか，または半角文字の “.,!?␣” (空白文字が入っていることに注意) のいずれかの文字が文の区切りとなる．

3 茶筌ライブラリ

茶筌ライブラリ lib/libchasen.a を利用することで，茶筌のモジュールを他のプログラムに組み込むことができる．利用できるライブラリ関数には以下のものがある．

```
int chasen_getopt_argv(char **argv, FILE *fp);

extern int Cha_optind;
```

茶筌にオプションを渡す．もし茶筌の初期化が行われていなければ，初期化を行ってからオプションの設定を行う．

茶筌ライブラリではスタンドアロンによる解析のみができる．-s,-D などサーバやクライアントに関するオプションは利用できない．

argv にはコマンドラインオプションとして NULL で終わる文字列の配列を指定する．ただし argv[0] はプログラムのファイル名である．オプション指定に誤りがあった場合，ファイル・ポインタ fp にエラーメッセージを出力する．ただし fp が NULL の時は何も出力しない．

オプション指定に誤りがなければ 0 を，誤りがあれば 1 を返す．

外部変数 Cha_optind には処理したオプション (argv[0] を含む) の数が格納される．

以下に使用例を示す．chawan というプログラムにおいて，'-r /home/rikyu/chasenrc.proj -j' というオプションを茶筌に渡している．この関数の実行後 Cha_optind には 4 が代入される．

```
char *option[] = {"chawan", "-r", "/home/rikyu/chasenrc.proj", "-j", NULL};
chasen_getopt_argv(option, stderr);
```

```
int chasen_fparse(FILE *fp_in, *fp_out);
```

```
int chasen_sparse(char *str_in, FILE *fp_out);
```

```
char *chasen_fparse_tostr(FILE *fp_in);
```

```
char *chasen_sparse_tostr(char *str_in);
```

スタンドアロンでの形態素解析を行う．もし茶筌の初期化が行われていなければ，初期化を行ってから形態素解析を行う．入力と出力がファイルであるか文字列であるかによって，4 つの関数がある．

chasen_fparse(), chasen_fparse_tostr() はファイル・ポインタ fp_in から読み込んだ文字列を解析する．文字コードとしては日本語 EUC あるいは JIS(ISO-2022-JP) を受け付ける．chasen_getopt_argv() で -j オプションを指定したときは，句点などを文の区切りとして解析を行う．

chasen_sparse(), chasen_sparse_tostr() は文字列 str_in を解析する．文字コードとしては日本語 EUC あるいは JIS(ISO-2022-JP) を受け付ける．

chasen_fparse(), chasen_sparse() は解析結果をファイル・ポインタ fp_out に出力する．コマンドモードで '#q' を実行して茶筌を終了したときは 1 を，それ以外のときは 0 を返す．

chasen_fparse_tostr(), chasen_sparse_tostr() は解析結果を茶筌内部で確保したメモリ領域に格納し，そのポインタを返す．この領域は，次に chasen_fparse_tostr(), chasen_sparse_tostr() を呼び出すまで有効である．コマンドモードで '#q' を実行して茶筌を終了したときは NULL を返す．

4 他のシステムからの利用

4.1 Prolog からの使用

- 基本的構成

  茶筌システムが形態素解析を行なった結果を，SICStus Prolog がパイプを介して受けとるという構成になっている．

- 動作環境

  茶筌が仮定する Prolog 処理系は，SICStus Prolog Release 3 である．日本語を用いるためには，環境変数 SP_CTYPE の値を euc に設定する必要がある．具体的には，例えば次の一行を利用者の.login ファイルなどに含めればよい．

```
setenv SP_CTYPE euc
```

- 起動方法

  Prolog を起動し，‘prolog/chasen_user.pl’ を consult もしくは compile する．

  ‘prolog/chasen_user.pl’ はユーザ設定用ファイルなので，ユーザが各自のディレクトリにコピーして各自の設定を行なうことを前提としている．

  解析は，以下の要領でファイル単位で行なうことができる．

```
| ?- cha.
Input file name?  入力ファイル名.
Output file name? 出力ファイル名.
```

  入力ファイル名，出力ファイル名を ‘user’ とすると，端末からの入出力が可能である．

```
| ?- chatty.
```

  とすれば，標準入力から入力し，標準出力へ結果を出力することができる．

- 処理内容

  Prolog 側からの一回の呼び出しによって，茶筌システムとのパイプがオープンされる．呼び出しが終了するとパイプが閉じられる．Prolog 側は，茶筌システムが形態素解析を行なった結果を受けとるだけである．一つの形態素について，Prolog 側が受けとるデータは，

  > morph([識別子 (ID), 開始位置 (From), 終了位置 (To), コスト (Cost), 見出し語 (Md), 読み (Ym), 基本形 (Kh), 品詞名 (Hn0), 品詞細分類 (Hn), 活用型名 (KT), 活用形名 (KF), 意味情報 (Imi), 形態素コスト (MrphCost), 前節形態素との接続コストのリスト (PreCCL), 前接形態素の識別子のリスト (PreIDL)])

  という複合項である．また，茶筌システムからデータを受けとる処理は，‘prolog/chasen.pl’ 中の cha/3 において，read(+*Str*,-*MorphList*) によって上述の複合項のリスト *MorphList* を読み込むという形で行なわれている．

  Prolog 版の出力オプションは，

  > e：完全な形態素情報を文字で表示 (デフォルト)
  > f：カラムを整えて表示
  > s：見出しと品詞名を表示

  であり，述語 cha_print_form/1 により変更することができる．

4.2 Perl からの使用

perl/ChaSen.pm を使うことにより，perl から茶筌を利用できる．インストール方法，使用方法については perl/README を参照のこと．

4.3 Emacs からの使用

茶筌クライアントの Emacs Lisp 版インタフェース chasen.el を使うことにより，Emacs 上で形態素解析を行うことができる．chasen.el を利用するには chasen/chasen.el をインストール (例えば /usr/local/lib/mule/site-lisp/ にコピー) し，.emacs 中で茶筌サーバのホスト名とポート番号の設定，autoload の指定を行う．

```
(setq chasen-server-host "kyusu")
(setq chasen-server-port 31234)    ; デフォルトは 31000

(autoload 'chasen-region "chasen" "ChaSen client" t)
(autoload 'chasen-line "chasen" "ChaSen client" t)
(autoload 'chasen-highlight-class-region "chasen" "ChaSen client" t)
(autoload 'chasen-property-class-region "chasen" "ChaSen client" t)
```

それぞれの関数の詳細については chasen.el の先頭のコメント部分を参照のこと．

参考文献

[1] 益岡隆志，田窪行則：『基礎日本語文法 –改訂版–』くろしお出版，1992.

[2] 妙木裕，松本裕治，長尾眞：「汎用日本語辞書および形態素解析システム」情報処理学会第 42 回全国大会予稿集，1991.

[3] 松本裕治，黒橋禎夫，宇津呂武仁，妙木裕，長尾真： 日本語形態素解析システム JUMAN 使用説明書 version 2.0，NAIST Technical Report, NAIST-IS-TR94025，1994.

[4] 北内 啓，山下 達雄，松本 裕治：「日本語形態素解析システムへの可変長連接規則の実装」, 言語処理学会第三回年次大会論文集, pp.437-440, 1997.

[5] 研究開発用知的資源タグ付きテキストコーパス報告書 平成 9 年度, テキストサブワーキンググループ, 技術研究組合 新情報処理開発機構, 1998.

付録

1 著作権および使用条件について

茶筌システムは，広く自然言語処理研究に資するため無償のソフトウェアとして開発されたものである．茶筌の著作権は，奈良先端科学技術大学院大学情報科学研究科自然言語処理学講座 (松本研究室) が保持する．本ソフトウェアの使用，改変，再配布については，特に制限を課すことはしないが，再配布については，次の事項を条件とする．

- 本ソフトウェアがその原形あるいは修正された形で再配布される場合は，再配布されるソフトウェアに茶筌 2.0 が使用されていることを明記すること．
- 再配布されるソフトウェアに，著作権に関する本節の記述と使用説明書の表紙裏のページの著作権に関する但し書きを必ず含むこと．

なお，本ソフトウェアの著作権者である奈良先端科学技術大学院大学は，原形あるいは改変された形で配布された本ソフトウェアに関連して生じる一切の損失に対して保証の責を負わないこととする．

2 JUMAN 2.0 から 茶筌 2.0 への拡張点

2.1 bi-gram 版と v-gram 版の相違点

茶筌 2.0 では品詞体系や連接規則の機能などを拡張した．この機能拡張版を v-gram 版，従来のバージョンを bi-gram 版と呼ぶ．v-gram 版は bi-gram 版と文法ファイルの形式が異なっているため，辞書に互換性がない．ただし，`mkchadic/convdic` を実行することにより，bi-gram 版の辞書を v-gram 版の辞書に変換することができる．

`convdic` は bi-gram 版の辞書があるディレクトリ上で，v-gram 版の辞書を格納するディレクトリを引数として実行する．例えば以下のように実行すると，bi-gram 版の辞書がある `dic1` というディレクトリと同じ階層に `dic2` というディレクトリが作成され，その中に v-gram 版の辞書が格納される．なお，`convdic` 実行後，茶筌に付属の `dic/Makefile` を v-gram 版の辞書があるディレクトリ (下の例では `dic2`) にコピーする必要がある．また，chasenrc ファイルも用意する．

```
% cd dic1
% ../mkchadic/convdic ../dic2
```

茶筌 2.0 ではデフォルトで v-gram 版がコンパイルされる．‘`make bigram`’ を実行すれば bi-gram 版の実行ファイルが作成され，bi-gram 版の辞書を利用することができる．

v-gram 版は bi-gram 版と比べ，以下のような拡張機能や変更点がある．

1. 品詞を 2 階層から多階層に拡張した．
2. 連接規則を bi-gram の固定長から variable-gram(可変長) に拡張した．すなわち，連接する 2 個の単語 (あるいは品詞) の連接コストだけではなく，3 個以上の任意の長さの単語 (品詞) 列に対して単語 (品詞) の連接コストを記述できる．
3. `*.dic` で「発音」という属性を使える．出力フォーマットの`%a`, `%A` で表示できる．また，`cforms.cha` で発音の語尾を定義できる．
4. `*.dic` で「base」という属性を使える．見出し語の基本形などを表示する際，活用を持っていればその基本形を，活用がなく base を持っていれば base を表示する．英語の辞書などで使用する．

5. chasenrc ファイルの「連結品詞」の機能を拡張し，複数の種類の品詞を別々に連結できるようにした．
6. 空行に対しても “EOS”(正確には BOS 文字列と EOS 文字列) を表示する．つまり，“EOS” の個数が入力文の行数と一致する．
7. 解析結果のデフォルトの出力形式 (-f) で，見出し語などの直後の区切りがスペースではなくタブになった．
8. 辞書に登録されていない単語の品詞表示を「未定義語」から「未知語」に変更した．
9. 形態素辞書ファイル *.dic で単語のコスト値が省略されている場合，bi-gram 版ではコスト値が 10 となるのに対し，v-gram 版では*.dic 中の「デフォルト品詞コスト」で指定されたコスト値 (指定されていない場合は 65535) が用いられる．
10. bi-gram 版では形態素コストと連接コストを内部で 10 倍しているが，v-gram 版ではそのままの値を用いる．また，bi-gram 版では形態素コストの範囲が 0 ~ 6553.5(茶筌 1.51 以前は 0 ~ 25.5) であるが，v-gram 版では 0 ~ 65535 である．
11. 連接コスト 0 を「確率 1 で連接する」という意味に，-1 を「連接しない」という意味に変更した．また，連接コストの範囲を -1 ~ 32767 に変更した．
12. 文節区切りの機能を持つ，長さ 0 の品詞が使える．品詞定義ファイルで品詞名の後ろに ‘/’ をつけると文節区切りとして機能する．

2.2 茶筌 1.5 から 茶筌 2.0 への拡張点

ここでは v-gram 版，bi-gram 版に共通する拡張点をあげる．

1. chasenrc の「文法ファイル」を省略できるようにした．「PATDIC」「SUFDIC」が ’/’ で始まっていない場合は，「文法ファイル」のディレクトリからの相対パスとみなすようにした．
2. 辞書引きに SUFARY を使えるようにすることにより，半角文字も検索できるようにした．
3. SUFARY を使って英語を解析できるようにした．
4. -D なしで -R を指定した場合は Makefile で指定した chasenrc (/usr/local/lib/chasen/dic/chasenrc など) を読み込むようにした．
5. 文頭・文末で出力する文字列を設定できるようにした．
6. 未知語品詞とそのコストを複数指定できるようにした．
7. chasenrc ファイルで「空白品詞」を指定することにより，空白も解析結果に出力できるようにした．
8. chasenrc ファイルで「注釈」を指定することにより、SGML タグのような特定の文字列を空白と同様に無視して解析できるようにした。
9. -lp, -lt, -lf オプションで品詞や活用のリストを表示できるようにした．
10. -o オプションで出力ファイルを指定できるようにした．
11. 出力フォーマット "%?T/STR1/STR2/" を使えるようにした．活用があれば STR1, なければ STR2 を出力する．そのほかに %?I, %?B, %?F, %?U も使えるようにした．
12. 出力フォーマット "%rABC" を導入し，ルビを表示できるようにした．
13. chasenrc ファイルで「BOS 文字列」「EOS 文字列」を指定することにより、文頭・文末で出力する文字列を設定できるようにした。
14. BOS 文字列，EOS 文字列，出力フォーマットで，解析文全体を表示する"%S" を使えるようにした．
15. 辞書ファイルの形態素コストの範囲を今までの 0 ~ 25.5 から，bi-gram 版は 0 ~ 6553.5 に，v-gram 版は 0 ~ 65535 に変更した．
16. 連接ファイルの連接コストの範囲を 0 ~ 255 から 0 ~ 32767 に変更した．

2.3 茶筌 1.0 から 茶筌 1.5 への拡張点

1. ライブラリ化を行い，茶筌のモジュールを他のプログラムに簡単に組み込めるようにした．

2. サーバ化を行い，クライアントを用いて他のマシンから解析を行うことができるようにした．また，クライアントの Emacs Lisp 版インタフェースを作成した．
3. -w オプションでコスト幅を指定できるようにした．
4. chasenrc ファイルに「区切り文字」を指定することにより，jfgets() の区切り文字を設定できるようにした．半角文字を指定することも可能．また，区切り文字のデフォルトを ” 。!? ” に変更した．
5. バッファを動的に確保することにより，文字列が長いときでも “Too many morphs” の警告が出ないようにした．
6. 美茶 (ViCha) 用出力オプション -v を新設した．
7. -d オプションと -b を同時に指定したときに-d の出力形式で最適解パスだけ表示できるようにした．

2.4 JUMAN 2.0 から 茶筌 1.0 への拡張点

1. 辞書検索の方法を従来の NDBM を用いて疑似的に TRIE 構造を実現する方法から，独自開発のパトリシア木を用いたものに変更した．解析に必要な辞書のサイズが約 4 分の 1 に縮小した．また，辞書のコンパイル時間が 3 ~ 40 分の 1 になった．
2. 解析システムの見直しを行ない，高速化を図った．解析速度が約 8 ~ 11 倍になった (JUMAN 2.0 との比較)．
3. 多くのプラットフォームでインストール可能になるようにコードを書き直した．また，GNU C コンパイラ (gcc) だけでなく OS 付属の C コンパイラなどでもコンパイルできるようにした．
4. 日本語 EUC だけでなく，JIS(ISO-2022-JP) の文字列も解析できるようにした．
5. 未定義連接コストの導入により，未定義語の出力を減らすことができるようになった．
6. 連結品詞を定義できるようにし，最適パスを出力する時に，その品詞の単語を一単語に連結して表示するようにした．
7. 活用語尾の読みを定義できるようにすることにより，「来る」「得る」などの読みがひらがなで表示されるようになった．
8. 入力文を改行コードで区切るのではなく，句点により区切るオプション (-j) を追加した．
9. -r オプションや環境変数 CHASENRC で chasenrc ファイルを指定できるようにした．
10. -F オプションや chasenrc ファイルの「出力フォーマット」で解析結果の出力形式を変更できるようにした．
11. 文法の見直しを行ない，品詞分類「特殊」の下の「括弧」を「括弧開」と「括弧閉」に分離した．また，同じく「特殊」の下に「空白」を定義した．「空白」は具体的には全角の空白を表す．
12. 助動詞の活用型に「助動詞べきだ型」を追加した．助動詞「べきだ」の活用を従来の「ナ形容詞」型から「助動詞べきだ型」に変更した．
13. 辞書登録語について見直し，追加削除等の修正を行なった．

3 JUMAN3.0 と 茶筌 との関係について

JUMAN 2.0 が 1994 年 7 月にリリースされて以降，京都大学長尾研究室と奈良先端大松本研究室では，それぞれ異なる方向での拡張を試みていました．京都大学では，従来の bi-gram モデルでは記述できない連接関係を記述するために連語処理や括弧の透過処理などの機能を追加し，文法ファイル，形態素辞書に大幅な修正を行なった拡張版を作成していました．奈良先端大では，今後大量の蓄積が始まると思われる日本語タグ付きコーパスから bi-gram 以上の連接規則 (単語レベルや品詞レベルの設定も含む) を自動的に学習する機能を追加するための拡張と，UNIX のハッシュデータベース NDBM に依存しない辞書の構築を考えていました．後者の拡張は UNIX 以外の OS での稼働を要求する声に対応することと辞書のコンパイル時間と検索速度の改善を目指したことによります．bi-gram 以上の連接規則に対する両者の考え方がかなり異なるため，両者の融合は

見合わせることにし，いち早く完成した京都大学の拡張版が 1996 年 6 月に JUMAN3.0beta として公開されました．

奈良先端大で拡張を予定していた機能には下に示すような項目があり，茶筌 1.0 を 1997 年 2 月に公開し，以後，茶筌 1.5, 1.51 を経て，茶筌 2.0b6 においてそのほとんどが実現されました．

1. (茶筌 1.0) 辞書システムの独自開発 (NDBM の棄却, パトリシア木の採用)
2. (茶筌 1.0) 解析システムの見直しと高速化
3. (茶筌 1.0) 未定義連接コスト，連接品詞，解析結果出力フォーマットの導入
4. (茶筌 1.0) JIS 文字列の解析
5. (茶筌 1.0) 活用語尾の読みの定義
6. (WinCha1.0) Windows への対応
7. (茶筌 1.5) ライブラリ化
8. (茶筌 1.5) サーバ化
9. (茶筌 2.0) 品詞定義の多階層化
10. (茶筌 2.0) 接続規則の可変長化
11. (茶筌 2.0) 半角文字を含む単語の辞書登録 (SUFARY を利用した辞書)
12. (茶筌 2.0) 出力フォーマットの拡充
13. 解析済みデータからの可変長接続コストの学習

4 添付の日本語辞書 (ipadic2.0) の品詞体系について

ここでは、茶筌 2.0 に添付の日本語辞書 (ipadic2.0) で採用した品詞体系について説明する。これは、情報処理振興事業協会 (IPA) で設定された IPA 品詞体系 (THiMCO97) に基づいて一部修正を加えたものである。本付録は新情報処理開発機構 (RWCP) による「テキストデータベース報告書 (平成 8 年度)」に掲載された IPA 品詞体系 (THiMCO97) の説明を許可を得て抜粋し、一部修正を施したものである。

なお、現在の IPA 品詞体系日本語辞書 (ipadic2.0) は、1998 年 5 月に公開した IPA 品詞体系日本語辞書 (ipadic1.0b2) に対して、奈良先端科学技術大学院大学情報科学研究科鹿野清宏教授を代表とする「日本語ディクテーション基本ソフトウェアの開発」(IPA 独創的先進的情報技術に関わる研究開発) のグループの方々に大幅な修正、改良を行っていただいたものである。

説明の書式

品詞名

以後、品詞名のことを「タグ」と呼ぶことがある。それぞれの品詞の説明の際に、以下の記号により注釈が付けられている。

＃ 品詞の解説

例： 単語例

＊ 品詞の解説についての備考

＆ 読み、活用形についての備考

品詞名に関する注意事項

本日本語辞書は、IPA 品詞体系 (THiMCO97) に基づいているが、茶筌の辞書として組み込む際にいくつかの変更を行なった。品詞体系の特徴と変更点について以下にまとめる。

- 品詞は、多段の階層に分類されている。例えば、「名詞 固有名詞 人名 姓」は、四段の階層よりなる品詞名である。以下では、これを「名詞-固有名詞-人名-姓」のようにハイフンで区切って表示する。茶筌 2.0 では、任意の段数の品詞階層の定義が可能になったので、これを直接文法ファイル (grammar.cha) に定義することができる。

- THiMCO97 では、「動詞 一段 連用形 自立」のように、品詞の分類と活用型、活用形が混ざり合った形で定義されていた。茶筌では、品詞の分類の定義と活用に関する定義が分離されているので、これを「動詞-自立 一段 連用形」のように 3 つの項目 (品詞名、活用型、活用形) に分けて記述することにした。

- 品詞名の定義に用いられる分類名を以下の基準に従って変更した。
  1. 「(助動詞語幹)」「(形容動詞語幹)」のように丸括弧を伴う名称の丸括弧をすべて除去した。
  2. 「動詞 接尾 (助動詞)」「形容詞 接尾 (助動詞)」として定義される「(助動詞)」の部分は冗長であるので、省略し、「動詞-接尾」「形容詞-接尾」とした。
  3. 動詞の分類には「動詞」「動詞 非自立」「動詞 接尾」に大別されるが、茶筌の品詞階層の定義では、「動詞」という記述はすべての動詞を表すので、区別のため、「動詞-自立」「動詞-非自立」「動詞-接尾」のように「自立」という細分類を追加した。同様に、活用語以外の単語のための品詞名については、「名詞」「名詞 固有名詞」「名詞 固有名詞 人名」「名詞 固有名詞 人名 姓」のような分類を、それぞれ、「名詞-一般」「名詞-固有名詞-一般」「名詞-固有名詞-人名-一般」「名詞-固有名詞-人名-姓」のように、「一般」という細分類を追加して、排他的に品詞の定義を行なった。
  4. 用言の活用形については、「未然ナイ接続」「未然レル接続」「未然ウ接続」「連用タ接続」「連用マス接続」「連用タイ接続」··· のように、後続する助動詞類に応じて細かく定義されていたが、個々の活用型については、「未然」「連用」等で異なる語尾形を持つものは少ない。よって、活用形の名称は、「未然形」「連用形」「基本形」「仮定形」「命令」を基本的な活用形とし、例外的な形のものに対してのみ、THiMCO97 の活用形名を使用した。なお、茶筌では辞書出現形に対して「基本形」という活用形を与える仕様になっているため、THiMCO97 の「見出し形」という活用形名を「基本形」という名前に変更した。
  5. 「未然ウ接続」は、五段活用の動詞については「う」が接続し、その他の活用型の動詞については「よう」が接続するための活用である。しかし、「う」「よう」を単語 (助動詞) として認めず、「未然ウ接続」については、「う」あるいは「よう」がついた形を活用形として「意志形」という名称を与えた。

- ipadic2.0 に登録の単語には新たに「発音」フィールドが追加された。これは、「日本語ディクテーション基本ソフトウェアの開発」グループの努力により添付されたものである。例えば、係助詞の「は」の読みは「ワ」、「常識」の読みは「ジョーシキ」のように長音は「ー」によって示されている。また、綴りも品詞も等しいが読みだけが異なる単語、例えば、「私 (ワタシ/ワタクシ)」については、{ ワタシ/ワタクシ } のようにすべての可能な読みを付与し、一つの語として登録した。

4.1 名詞

名詞-一般

普通名詞、あるいは、下位分類が未定の名詞。

名詞-固有名詞-一般

＃ 一般的な固有名詞、あるいは、下位分類が未定の固有名詞。

例：「関西国際空港」

名詞-固有名詞-人名-一般

＃ 姓と名に分けられないもの、外国人名。あるいは、姓・名の決定が未定の人名。

例：「Ａ・Ｇ・スポルディング」

名詞-固有名詞-人名-姓

＃ 主に日本人の姓。

例：「山田」...

名詞-固有名詞-人名-名

＃ 主に日本人の名。

例：「太郎」...

名詞-固有名詞-組織

＃ 組織を表わす名称。

例：「通産省」「ＮＨＫ」...

名詞-固有名詞-地域-一般

＃ 国名以外の地名を表わすもの。

例：「アジア」「スマトラ島」

名詞-固有名詞-地域-国

＃ 国の名前。

例：「日本」「オーストラリア」...

名詞-代名詞-一般

＃ いわゆる代名詞、不定語。

例：「それ」「ここ」「あいつ」「あなた」「あちこち」「いくつ」「どこか」「なに」「みなさん」「みんな」「わたくし」「われわれ」...

名詞-代名詞-縮約

代名詞と係助詞「は」の組み合わせで、短縮した形 < 口語 > 。

例：「ありゃ」「こりゃ」「こりゃあ」「そりゃ」「そりゃあ」

名詞-副詞可能

曜日、月など時間を表す副詞的な用法を持つ名詞。量や割合などを表し副詞的に使うことのできる名詞。

例：「金曜」「一月」「午後」「少量」...

* 元の IPA 品詞体系では、「名詞-副詞可能」のうち実際に副詞的に使われてるものを「名詞 副詞可能 副詞的」、副詞的な使用が可能であるが、副詞的に用いられていないものを「名詞 副詞可能」とラベル付けされることになっているが、ここでは、文内の用法に関係なく副詞的に働き得るものをすべて「名詞-副詞可能」と呼ぶ。

名詞-サ変接続

格要素をとり、後ろに「する」「できる」「なさる」「くださる」「、」「。」が後接することができるもの。

例：「インプット」「愛着」「悪化」「悪戦苦闘」「一安心」「下取り」「具体化」...

* 「オノマトペ (+ スル)」は、[副詞-助詞類接続] とした。

名詞-形容動詞語幹

いわゆる形容動詞語幹で、「な」の前に現れるもの。

例：「健康」「安易」「駄目」「だめ」...

* 元の IPA 品詞体系では「名詞 (形容動詞語幹)」となっていたが、第 2 階層の「(形容動詞語幹)」の括弧を取り除いた。

名詞-ナイ形容詞語幹

助動詞の「ない」の直前に現れて形容詞的な働きをする語

例：「申し訳」「仕方」「とんでも」「違い」...

* 元の IPA 品詞体系では形容詞とみなされていたが、「申し訳-ない」「申し訳-ありません」「申し訳-ございません」のように派生するので、語幹として統一的に扱うことにした。ただし、「ナイ形容詞語幹」として分類された語がすべてこのような用法を持つわけではない。

名詞-数

漢数字、算用数字、および、「何 (回)」「数 (%)」「幾 (夜)」。

例：「０」「１」「２」「何」「数」「幾」..

名詞-非自立-一般

連体詞、「の (格助詞)」、活用語の連体形 [見出し形] に接続して使われるもののうち、以下の下位分類にあてはまらないもの。いわゆる形式名詞を含む。

* 普通名詞としての用法であれば、連体修飾を受けていても [非自立] ではない。

例：「あかつき」「暁」「かい」「甲斐」「気」「きらい」「嫌い」「くせ」「癖」「こと」「事」「ごと」「毎」「しだい」「次第」「順」「せい」「所為」「ついで」「序で」「つもり」「積もり」「点」「どころ」「の」「はず」「筈」「はずみ」「弾み」「拍子」「ふう」「ふり」「振り」「ほう」「方」「旨」「もの」「物」「者」「ゆえ」「故」「ゆえん」「所以」「わけ」「訳」「わり」「割り」「割」「ん < 口語 > 」「もん < 口語 > 」...

名詞-非自立-副詞可能

連体詞、「の (格助詞)」、活用語の連体形 [見出し形] に接続して使われるもののうち、副詞的に働くことが可能なもの。

* 文脈上で、実際に副詞的に働いている場合には、IPA 品詞体系ではうしろに [副詞的] を付加することになっているが、それは省略した。

例：「あいだ」「間」「あげく」「挙げ句」「あと」「後」「余り」「以外」「以降」「以後」「以上」「以前」「一方」「うえ」「上」「うち」「内」「おり」「折り」「かぎり」「限り」「きり」「っきり」「結果」「ころ」「頃」「さい」「際」「最中」「さなか」「最中」「じたい」「自体」「たび」「度」「ため」「為」「つど」「都度」「とおり」「通り」「とき」「時」「ところ」「所」「とたん」「途端」「なか」「中」「のち」「後」「ばあい」「場合」「日」「ぶん」「分」「ほか」「他」「まえ」「前」「まま」「儘」「侭」「みぎり」「矢先」...

名詞-非自立-形容動詞語幹

連体詞、「の (格助詞)」、活用語の連体形 [見出し形] に接続して使われるもののうち、「な (助動詞「だ」) の体言接続」と接続可能なもの。

例：「みたい」「ふう」「よう」

* 元の IPA 体系では、「名詞 非自立 (形容動詞語幹)」と書かれている。元の IPA 体系では、「よう」が「名詞 非自立 (助動詞語幹)」として区別されていたが、特に区別する理由がないので、「よう」も「名詞-非自立-形容動詞語幹」とした。

*

名詞-特殊-助動詞語幹

終止形 [見出し形] に接続するもので、学校文法で助動詞とされている「そうだ (伝聞)」の語幹部分。

例：「そう」

* 元の IPA 体系では、「名詞 特殊 (助動詞語幹)」と書かれている。

名詞-接尾-一般

名詞、あるいは他の品詞の語幹 [ガル接続] や [連用タイ接続] に接続して複合名詞を形成する語のうち、下位の分類にあてはまらないもの。一般に「接尾語」というよりも範囲が広く、複合名詞の後ろ要素として用いられることが普通なもの。

例：「おき」「かた」「方」「甲斐 (がい)」「がかり」「ぎみ」「気味」「ぐるみ」「(~ した) さ」「次第」「済 (ず) み」「よう」「(でき) っこ」「感」「観」「性」「学」「類」「面」「用」...

名詞-接尾-サ変接続

名詞に接続して名詞を形成する接尾語のうち「スル」に前接し得るもの。

例：「化」「視」「分け」「入り」「落ち」「買い」

名詞-接尾-助動詞語幹

他の品詞の連用形に接続し、学校文法で助動詞の語幹とされている「そうだ (様態)」の語幹部分。

例：「そう」

* 元の IPA 体系では、「名詞 接尾 (助動詞語幹)」と書かれている。

名詞-接尾-形容動詞語幹

他の名詞や活用語の連用形に接続する接尾語で、「な」([助動詞- 特殊型-体言接続]) に前接するもの。

例：「的」「げ」「がち」

* 元の IPA 体系では、「名詞 接尾 (形容動詞語幹)」と書かれている。

名詞-接尾- 副詞可能

他の名詞に接続する接尾語で、副詞的に働くことが可能なもの。

* IPA 品詞体系では、文脈上、実際に副詞的に働いているものは、うしろに [副詞的] と付加してあるが、ここでは具体的な用法にかかわらず、副詞的な使用が可能なものをすべてこの分類とした。

例：「後 (ご)」「以後」「以降」「以前」「前後」「中」「末」「上」「時 (じ)」

名詞-接尾-助数詞

数に接続して名詞を形成する接尾。一般の「助数詞」よりも範囲が広く、数に接続する普通名詞も含まれる。

例：「個」「つ」「本」「冊」「パーセント」「ｃｍ」「ｋｇ」「カ月」「か国」「区画」「時間」「時半」...

* IPA 品詞体系では、これらのうち副詞的に用いられているものに「名詞 接尾 助数詞 副詞的」というタグを与えているが、これは用法に関するタグであるため、本体系では含めなかった。

名詞-接尾-特殊

＃ 主に用言につく特殊な接尾辞として新たに定義した。

例：「(楽し) さ」「(考え) 方」

＊ IPA 品詞体系では、「名詞 接尾」に分類されている。

名詞-接続詞的

＃ 単語と単語を接続する接続詞的な働きをするもの。

例：「(日本) 対 (アメリカ)」「対 (アメリカ)」「(３) 対 (５)」「(女優) 兼 (主婦)」

名詞-動詞非自立的

＃ [助詞-接続助詞] の「て」に接続するもので、意味的には動詞的なもの。

例：「ごらん」「ご覧」「御覧」「頂戴」

注 IPA 品詞体系には、単語への分割が不可能なもの、および、ことわざ、漢詩、方言、英語などを表すタグとして「名詞 引用文字列」が用意されている。また、数式を表すためのタグ「名詞 数式」が用意されている。これらは品詞タグとは考えにくいため、本体系では含めなかった。

4.2 接頭詞

接頭詞-名詞接続

＃ 名詞 (形容動詞語幹を含む) に前接する接頭語のうち、数に接続するもの以外。

例：「お (水)」「某 (氏)」「同 (社)」「故 (~ 氏)」「高 (品質)」「お (見事)」「ご (立派)」

接頭詞-数接続

＃ 名詞に前接する接頭語のうち、数に接続するもの。

例：「約」「全長」「弱冠」「月 (千円)」「年 (一回)」

接頭詞-動詞接続

＃ 動詞の命令形あるいは [動詞 連用タイ接続] +「なる / なさる / くださる」に前接する接頭語。

例：「お (読みなさい)」「お (座り)」

接頭詞-形容詞接続

＃ 形容詞に前接する接頭語。

例：「お (寒いですねえ)」「バカ (でかい)」

4.3 動詞

活用形に関する注意

未然形 THiMCO97 では以下のように細かく分類されているが、語尾形に変化のない限り、「未然形」に統一した。なお、「未然ウ接続」については、「う」あるいは「よう」がついた形を活用形として「意志形」という名称を与えた。

- 未然レル接続

  ＃ −(ラ)レル,−(サ)セルに接続するもの。

  例：「読ま」「さ」...

- 未然ナイ接続

  ＃ −ナイに接続するもの。

  例：「読ま」「し」...

- 未然ヌ接続

  ＃ −ヌ,−(サ)シメルに接続するもの。

  例：「読ま」「せ」「来」...

- 未然ウ接続

  ＃ −(ヨ)ウに接続するもの。

  例：「読も」「し」...

  ＆ ipadic1.0 以降では、(ヨ)ウが接続した形を「意志形」と定義した。

連用形 例外的な語尾以外はすべて「連用形」という名称に統一した。

- 連用マス接続

  ＃ −マスに接続するもの。

  例：「読み」「し」「なさい」...

- 連用タイ接続

  ＃ −タイ,−ソウ,−ヅライ,−方(かた),読点などに接続するもの。

  例：「読み」「し」「なさり」「向かひ」「習ひ」...

- 連用タ接続

  ＃ −タ,−テに接続するもの。

  例：「読ん」「書い」「行っ」「問う」...

基本形 THiMCO97 では、「見出し形」と呼ばれているもの。

＃ 句点,体言,−マイなどに接続するもの。

例：「読む」「なさる」「問う」...

仮定形 THiMCO97 では、「仮定バ接続」と呼ばれている。

＃ −バ,−ドモに接続するもの。

例：「読め」「すれ」...

命令ｉ ＃ カ変・五段ラ行特殊の命令形。およびサ変・スルの命令形「せよ」の口語形。

例：「来い」「なさい」「せい」...

命令ｅ　＃　五段の命令形、文語已然形、一段動詞の語幹止め命令用法 (「くれ」のみ)。

例：「読め」「(とは) いえ」「(程度の差こそ) あれ」「(やめて) くれ」...

＊　「(やめて) くれ」は「(やめて) くれろ」の「ろ」が落ちた形。「くれる」は一段動詞の中の特殊活用型とすべきものである。なお、「(やめて)(お) くれ (なさい)」の「くれ」は [動詞-非自立 一段 連用タイ接続] であり、別のものであり、この口語形は「おくんなさい」となる。

命令ｙｏ　＃　一段・サ変・文語 (カ変) の命令形で「ｙｏ」で終わるもの。

例：「せよ」「みよ」「来よ」...

命令ｒｏ　＃　一段・サ変の命令形で「ｒｏ」で終わるもの。

例：「しろ」「みろ」...

ベキ接続　＃　「ベキ」につづく形、サ変の場合のみ。

例：「す」...

仮定縮約１　＃　仮定バ接続と「バ」とが合わさって短縮した形 < 口語 > 。

例：「分かれりゃ」

体言接続　＃　文語の場合のみ。見出し形と異なる形があるもの。

例：「助くる」(cf.「助く」)

体言接続特殊　＃　「る」で終る動詞が「の」などに接続する場合に音便化した形 < 口語 > 。

例：「(何) すん (の？)」

動詞の活用型一覧 (現代語)

＃＃【活用形】の位置には次のものが入る。(【】の記号はない)

動詞-自立 カ変 【活用形】

例：「くる」「来る」「やってくる」「やって来る」

動詞-非自立 カ変 【活用形】

例：「(て) くる」「(て) 来る」

動詞-自立 サ変・スル 【活用形】

＃　「する」、および、[名詞 サ変接続] に接続する「する」。

例：「する」

動詞-非自立 サ変・スル 【活用形】

＃ 動詞の [連用タイ接続](＋ [助詞-係助詞]) に接続する「する」。

例：「(お読み) する」「(~ を読みも) する」「(~ を読みは) する」...

＊ 「お読みする」は、「お」は [接頭詞-動詞接続]、「読み」は [動詞 五段・マ行 連用タイ接続]、「する」は [動詞 サ変・スル 見出し形 非自立]。

＊ 「 ~ を読みもする」「 ~ を読みはする」は、「読み」は [動詞 五段・マ行 連用タイ接続]、「も」は [助詞 係助詞]、「する」は [動詞 サ変・スル 見出し形 非自立]。

動詞-自立 サ変・−スル 【活用形】

＃ 和語系のサ変動詞。

例：「接する」...

＊ 「−し＋ない」「−せ＋られる」「−せ＋ぬ」「−し＋よう」「−する」「−すれ＋ば」「−せよ」「−しろ」の形だけを [動詞 サ変・スル] とした。「−し＋、 」「−し＋た」「−し＋たい」などの連用形はすべて [動詞 五段・サ行] とした。

動詞-自立 サ変・−ズル 【活用形】

＃ 和語系のザ変動詞。

例：「信ずる」...

＊ 「−ぜ＋られる」「−ぜ＋ぬ」「−ずる」「−ずれ＋ば」「−ぜよ」「−ず＋べし」の形だけを [動詞 サ変・−ズル] とした。「−じ＋ない」「−じ＋よう」の未然形および「−じ＋、 」「−じ＋た」「−じ＋たい」などの連用形、および「−じろ」の命令形は [動詞 一段] とした。

動詞-自立 一段 【活用形】

＃ 上一段活用および下一段活用。

例：「着る」

＊ 「病める」は、[見出し形] のみ。

動詞-非自立　一段 【活用形】

例：「あげる」「うる」「える」「得る」「おえる」「終える」「おおせる」「かねる」「兼ねる」「かける」「きれる」「切れる」「すぎる」「過ぎる」「そこねる」「損ねる」「そびれる」「そめる」「初める」「つける」「つづける」「続ける」「(お読み) できる」「(お読み) 出来る」「はじめる」「始める」「(て) いる」「(~ しては) いけ (ない)」「(て) くれる」「(て) 差し上げる」「(て) のける」「(て) みる」「(て) みせる」[(て) もらえる」「(て) る < 口語 > 」

＊ 「(~ しては) いけ (ない)」の終止形は「いける」。

＊ 「(勉強) できる」は [非自立] としない。

＊ 「うる」は、「うる」([見出し形]) と「うれ」([仮定バ接続]) のみ。「う」は [動詞 文語 見出し形] とする。

動詞-接尾 一段 【活用形】

＃ 学校文法では助動詞と呼ばれているもの。

例：「させる」「せる」「しめる」「しむる」「られる」「れる」

動詞-自立 五段・カ行イ音便 【活用形】

＃ 五段カ行で、[助詞 接続助詞] の「て」に接続するときにイ音便になるもの。

例：「解く」...

動詞-非自立 五段・カ行イ音便 【活用形】

例：「つづく」「続く」「ぬく」「抜く」「(て) いただく」「(て) 頂く」「(て) おく」「とく < 口語 > 」「どく < 口語 > 」

動詞-非自立 五段・カ行促音便 【活用形】

＃ 五段カ行で、[助詞 接続助詞] の「て」に接続するときに促音便になるもの。

例：「いく」「行く」「ゆく」

＊ 「ゆく」には、「ゆっ (て)」の形はないが、このタグを振る。「ゆき (て)」は [動詞 文語 連用タ接続] とする。

動詞-非自立 五段・カ行促音便 【活用形】

例：「いく」「行く」「ゆく」「く < 口語 > 」

＊ 「ゆく」には、「ゆっ (て)」の形はないが、このタグを振る。「ゆき (て)」は [動詞 文語 連用タ接続] とする。

動詞-自立 五段・ガ行 【活用形】

＃ 五段ガ行で、[助詞 接続助詞] の「て」に接続するときにイ音便になるもの。

例：「継ぐ」「急ぐ」...

動詞-自立 五段・サ行 【活用形】

＃ 五段サ行で、[助詞 接続助詞] の「て」に接続するときに音便化しないもの。

例：「話す」...

動詞-非自立 五段・サ行 【活用形】

例：「いたす」「致す」「だす」「出す」「つくす」「尽くす」「直す」

動詞-自立 五段・タ行 【活用形】

＃ 五段タ行で、[助詞 接続助詞] の「て」に接続するときに促音便になるもの。

例：「持つ」...

動詞-自立 五段・ナ行 【活用形】

＃ 五段ナ行で、[助詞 接続助詞] の「て」に接続するときにハツ音便になるもの。

例：「死ぬ」

動詞-自立 五段・バ行 【活用形】

＃ 五段バ行で、[助詞 接続助詞] の「て」に接続するときにハツ音便になるもの。

例：「呼ぶ」...

動詞-自立 五段・マ行 【活用形】

＃ 五段マ行で、[助詞 接続助詞] の「て」に接続するときにハツ音便になるもの。

例：「進む」...

動詞-非自立 五段・マ行 【活用形】

例：「こむ」「込む」

動詞-自立 五段・ラ行 【活用形】

＃ 五段ラ行で、[助詞 接続助詞] の「て」に接続するときに促音便になるもの。

例：「切る」「なる」...

動詞-非自立 五段・ラ行 【活用形】

例：「おわる」「終る」「終わる」「かかる」「きる」「切る」「しぶる」「渋る」「まいる」「まわる」「回る」「やがる」「(せねば / しては) なら (ない)」「(て) ある」「(て) おる」「(て) まわる」「(て) 回る」「(て) やる」「ちゃる < 口語 > 」「じゃる < 口語 > 」「ぢゃる < 口語 > 」

＊ 「なら (ない)」の終止形は「なる」

動詞-接尾 五段・ラ行 【活用形】

例：「がる」

動詞-自立 五段・ラ行特殊 【活用形】

＃ 五段ラ行で、助動詞「ます」に接続する形および命令形が「-い」の形になるもの。

例：「いらっしゃる」「おっしゃる」「仰言る」「くださる」「下さる」「なさる」「ござる」

動詞-非自立 五段・ラ行特殊 【活用形】

例：「(お読み) なさる」「(お読み) くださる」「(お読み) 下さる」「(て) くださる」「(て) 下さる」「(て) いらっしゃる」「(て) らっしゃる < 口語 > 」

＊ 「(お読み) なさる」「(お読み) くださる」は [非自立] としたが、「(ご期待) なさる」「(ご期待) くださる」は [非自立] としない。

動詞-自立 五段・ワ行ウ音便 【活用形】

＃ 五段ワ行で、[助詞 接続助詞] の「て」に接続するときにウ音便になるもの。

例：「問う」「乞う」「沿う (て)」「ゆう (て)」「食う (て)」「すう (て)」「負う (て)」

＊ [動詞 五段・ワ行促音便] の動詞については、「て」に接続する活用語尾が「う」になっている場合のみ、このタグを振り、それ以外は、[動詞 五段・ワ行促音便] を振る。(人手修正データ中に現れたものは「ゆう」「食う」「すう」「負う」)

動詞-非自立 五段・ワ行ウ音便 【活用形】

例：「たまう」「給う」

動詞-自立 五段・ワ行促音便 【活用形】

＃ 五段ワ行で、[助詞 接続助詞] の「て」に接続するときに促音便になるもの。

例：「言う」「ゆう」「「食う」「負う」「憂う」..

＊ 「憂う」には「憂って」はないが、このタグを振る。(人手修正データ中に現れたものは「憂い (、)」の形のみ)

＊ [動詞 五段・ワ行促音便] の動詞については、「て」に接続する活用語尾が「う」になっている場合のみ、[動詞 五段・ワ行ウ音便] を振る。

動詞-非自立 五段・ワ行促音便 【活用形】

例：「あう」「合う」「そこなう」「損なう」「(て) しまう」「(て) もらう」「じゃう < 口語 > 」「じまう < 口語 > 」「ちまう < 口語 > 」「ちゃう < 口語 > 」

動詞の活用型一覧 (文語)

IPA 品詞体系では、文語の活用型の細分類は行われていない。現在の辞書は、文語動詞の活用型として次のものだけを記述した。今後充実させる必要がある。活用体系が文語残存のものと、口語ではあるが歴史仮名づかいで示されているものとを含む。

動詞-自立 文語・四段 【活用形】

例：「いふ」「云ふ」「向かふ」「習ふ」「思ふ」「能ふ」

＊ 「云へ (り)」は已然形だが、[動詞 文語・四段 命令ｅ] とした。

動詞-自立 文語・ラ変 【活用形】

例：「あり」「なり」「しかり」

動詞-自立 文語・下二 【活用形】

例：「用ゆ」

4.4 形容詞

「見出し形」「仮定バ接続」「文語見出し形」をそれぞれ「基本形」「仮定形」「文語基本形」と呼ぶ以外は、ほぼ THiMCO97 で用いられている活用形名を用いた。なお、形容詞の活用型を「形容詞・アウオ段」「形容詞・イ段」「形容詞・文語」に分類した。

未然ヌ接続

＃ −ヌに接続するもの。

例：「寒から」...

未然ウ接続

＃ −ウに接続するもの。

例：「寒かろ」...

連用タ接続

＃ −タに接続するもの。

例：「寒かっ」...

連用テ接続

＃ −テ，−ナイ，−ナル，−スル，読点に接続するもの。

例：「寒く」...

連用ゴザイ接続

＃ −ゴザイマスに接続するもの。

例：「寒う」「大きゅう」「のう」...

基本形

＃ 句点，体言などに接続するもの。

例：「寒い」「大きい」「ない」...

体言接続

＃ 文語活用で体言に接続するもの。

例：「寒き」「なき」...

＆ 基本形には「−イ」の形を入れた。

仮定形

＃ −バに接続するもの。

例：「寒けれ」「なけれ」...

＆ THiMCO97 では「仮定バ接続」と呼ばれていた。

命令

＃ 文語活用で命令形のもの。

例：「よかれ」「美しかれ」...

＆ 終止形には「−イ」の形を入れた。

ガル接続

＃ −ガル，−ゲ，−ソウに接続するもの。

例：「寒」「悲し」...

文語基本形

＃ −シで終わるもの。

例：「良し」「遠し」「やむなし」...

文語体言接続

＃ −キで終わって体言に接続するもの、「−イ」で終わる形のないもの。

例：「悪しき」...

仮定縮約 1

＃ 仮定バ接続と「バ」とが合わさって短縮した形 1 < 口語 > 。

例：「欲しけりゃ」「(それが) なけりゃ (困る)」

仮定縮約 2

＃ 仮定バ接続と「バ」とが合わさって短縮した形 2 < 口語 > 。

例：「(それが) なきゃ (困る)」

＃＃【活用形】の位置には次のものが入る。(【】の記号はない)

形容詞-自立 形容詞・アウオ段 【活用形】

＃ 形容詞の活用型のうち、語幹の最後の母音がアウオのいずれかで終わるもの。

例：「青い」「赤い」「厚い」「暑い」「熱い」...

＊ IPA 品詞体系では、「ない」の文語見出し形「なし」を形容詞の文語型活用の見出し形として定義しているが、本体系では、「形容詞・アウオ段 (あるいは、形容詞・イ段)」型の「文語見出し形」という活用形として定義している。同様に、IPA 体系で形容詞の文語型活用の体言接続と定義されている「悪しき」などは、本体系では「形容詞・文語」型の「文語体言接続」形として定義した。

形容詞-自立 形容詞・イ段 【活用形】

＃ 形容詞の活用型のうち、語幹の最後の母音がイで終わるもの。

例：「哀しい」「楽しい」「頼もしい」...

形容詞-自立 形容詞・文語 【活用形】

＃ 形容詞の中で「文語体言接続」の用法をもつもの。

例：「悪し」

形容詞-非自立 形容詞・アウオ段 【活用形】

＃ 動詞の [連用タイ接続] あるいは [連用タ接続] に後接する形容詞。

例：「がたい」「難い」「づらい」「にくい」「やすい」「(て) よい」「(て) 良い」

形容詞-非自立 形容詞・イ段 【活用形】

＃ 動詞の [連用タイ接続] あるいは [連用タ接続] に後接する形容詞。

例：「らしい」「(て) いい」「(て) ほしい」「(て) 欲しい」

形容詞-接尾 形容詞・アウオ段 【活用形】

＃ 学校文法では助動詞とされるもの。

例：「(食べ) たい」

4.5 副詞

副詞-一般

＃ 必ず後ろで切れるもの、連体修飾が不可能なもの。

例：「何だか」「何はともあれ」「何といっても」「何がなんだか」「何もかも」「何となく」「何とか」「何かと」「何より」「何とも (申し訳ないです)」「何しろ」「何でも (このパーティーは彼の主催だそうだ)」「なんら」「何かしら」「何やら」「何で」「何も (おこらなくったって)」「どんなに」「こんなにも」...

副詞-助詞類接続

＃ 「の」「は」「に」「な」「する」「だ」などが後続することが可能な副詞。

例：「こんなに」「そんなに」「あんなに」「なにか」「なんでも」

4.6 連体詞

連体詞

＃ 名詞を修飾する形しかもたないもの。

例：「この」「その」「あの」「どの」「いわゆる」「なんらかの」「何らかの」「いろんな」「こういう」「そういう」「ああいう」「どういう」「こんな」「そんな」「あんな」「どんな」「大きな」「小さな」「おかしな」「ほんの」「たいした」「(−も) さる (ことながら) 」「微々たる」「堂々たる」「単なる」「いかなる」「我が」「同じ」「亡き」...

4.7 接続詞

接続詞

＃ 独立に現れる接続詞。

例：「が」「けれども」「そして」「じゃあ」「それどころか」...

4.8 助詞

助詞-格助詞-一般

＃ いわゆる格助詞。

＊ 「にて」「とて」も格助詞に含めた。「の」には格助詞としての用法と名詞と名詞を接続するいわゆる「 A のＢ」用法があるが、今回は区別をせず格助詞として統一した。

例：「から」「が」「で」「と」「に」「へ」「より」「を」「の」「にて」「とて」

助詞-格助詞-引用

＃　名詞、人物の台詞「 」、会議の決定事項、理由、判決、推測表現等の直後の「と」。

例：「(~だ) と (述べた。)」「(~である) と (して執行猶予...)」「(なん) て (魚？) < 口語 > 」

助詞-格助詞-連語

＃　格助詞と動詞との連語で、主に格助詞に相当するような働きを持つもの。

例：「という」「といった」「とかいう」「として」「とともに」「と共に」「でもって」「にあたって」「に当たって」「に当って」「にあたり」「に当たり」「に当り」「に当たる」「にあたる」「において」「に於いて」「に於て」「における」「に於ける」「にかけ」「にかけて」「にかんし」「に関し」「にかんして」「に関して」「にかんする」「に関する」「に際し」「に際して」「にしたがい」「に従い」「に従う」「にしたがって」「に従って」「にたいし」「に対し」「にたいして」「に対して」「にたいする」「に対する」「について」「につき」「につけ」「につけて」「につれ」「につれて」「にとって」「にとり」「にまつわる」「によって」「に依って」「に因って」「により」「に依り」「に因り」「による」「に依る」「に因る」「にわたって」「にわたる」「をもって」「を以って」「を通じ」「を通じて」「を通して」「をめぐって」「をめぐり」「をめぐる」「って < 口語 > 」「ちゅう < 関西弁「という」 > 」「(何) ていう (人) < 口語 > 」「っていう < 口語 > 」「といふ」「とかいふ」

助詞-接続助詞

例：「から」「からには」「が」「けれど」「けれども」「けど」「し」「つつ」「て」「で」「と」「ところが」「どころか」「とも」「ども」「ながら」「なり」「ので」「のに」「ば」「ものの」「や (~した)」「やいなや」「(ころん) じゃ (いけない) < 口語 > 」「(行っ) ちゃ (いけない) < 口語 > 」「(言っ) たって (しかたがない) < 口語 > 」「(それがなく) ったって (平気) < 口語 > 」

助詞-係助詞

例：「こそ」「さえ」「しか」「すら」「は」「も」「ぞ」

助詞-副助詞

例：「がてら」「かも」「くらい」「位」「ぐらい」「しも」「(学校) じゃ (これが流行っている) < 口語 > 」「(それ) じゃあ (よくない) < 口語 > 」「ずつ」「(私) なぞ」「など」「(私) なり (に)」「(先生) なんか (大嫌い) < 口語 > 」「(私) なんぞ」「(先生) なんて (大嫌い) < 口語 > 」「のみ」「だけ」「(私) だって < 口語 > 」「だに」「(彼) ったら < 口語 > 」「(お茶) でも (いかが)」「等 (とう)」「(今後) とも」「ばかり」「ばっか < 口語 > 」「ばっかり < 口語 > 」「ほど」「程」「まで」「迄」「(誰) も (が)([助詞-格助詞] および [助詞-係助詞] の前に位置する「も」)

助詞-並立助詞

例：「と」「たり」「だの」「だり」「とか」「なり」「や」「やら」

助詞-終助詞

例：「かい」「かしら」「さ」「ぜ」「(だ) っけ < 口語 > 」「(とまってる) で < 方言 > 」「な」「ナ」「なあ < 口語 > 」「ぞ」「ね」「ネ」「ねぇ < 口語 > 」「ねえ < 口語 > 」「ねん < 方言 > 」「の」「のう < 口語 > 」「や」「よ」「ヨ」「よぉ < 口語 > 」「わ」「わい < 口語 > 」

＊ 終助詞の「や」は「(まあいい) や」「(すごい) や」など。関西方言の「や」は、不変化活用の助動詞として扱う。

助詞-副助詞 / 並立助詞 / 終助詞

＃ 「か」のうち、副助詞、並立助詞、終助詞いずれかわからないもの、たとえば、次の (a)(b)(c) のようなもの。

＃ (a)「ＡかＢか」型。例 : :「(国内で運用する) か、(海外で運用する) か (。)」

＃ (b) 副詞節中。例 :「(幸いという) か (、死者はいなかった。)」「(祈りが届いたせい) か (、試験に合格した。)」

＃ (c)「かのように」。例 :「(何もなかった) か (のように振る舞った。)」

例：「か」

＊ 最新の IPA 品詞体系では、これをさらに「副助詞」「並立助詞」「終助詞」に細分類しているが、本体系では区別しなかった。

助詞-連体化

＃ [名詞] に接続して体言にかかる「の」。

＊ THiMCO97 では、この用法の「の」も格助詞に分類されている。

助詞-副詞化

＃ 擬音語、擬声語、擬態語及びそれに類する名詞や副詞の直後の「に」「と」。

例：「に」「と」

＊ ただし、「する」「なる」に係っているもので、状態変化を表す場合は格助詞とする。

助詞-特殊

＃ 以上の分類にあてはまらないもの。短歌や俳句等に用いられる助詞などを含む。

例：「かな」「けむ」「(~ しただろう) に」「(あんた) にゃ (わからん)」「(俺) ん (家)」

4.9 助動詞

助動詞 五段・ラ行アル 【活用形】

＃ 動詞型の活用の助動詞。「である」「ではある」などの「ある」。

例：「ある」

助動詞 五段・ラ行ゴザル 【活用形】

＃ 動詞型の活用の助動詞「ござる」。

例：「ござる」

助動詞 形容詞・イ段 【活用形】

＃ 形容詞型活用の助動詞。

例：「らしい」

助動詞 特殊・ナイ 【活用形】

＃ 否定の助動詞「ない」の活用型。

例：「ない」

助動詞 特殊・タ 【活用形】

＃ 完了を表す助動詞「た」の活用型。

例：「た」「だ」

＊ 「(学ん) だ」「(泳い) だ」のように五段活用のガ行、ナ行、バ行、マ行に接続する場合には表層形が「だ」になるため、本体系では、それぞれ別の語幹をもつ形態素として定義した。

助動詞 特殊・ダ 【活用形】

＃ 断定の助動詞「だ」の活用型。

例：「だ」

助動詞 特殊・デス 【活用形】

＃ 断定の助動詞「です」の活用型。

例：「です」

助動詞 特殊・ジャ 【活用形】

＃ 断定の助動詞「じゃ」の活用型。

例：「じゃ」

＊ 断定の「だ」が鈍ったもの。

助動詞 特殊・マス 【活用形】

＃ 謙譲・丁寧を表わす助動詞「ます」の活用型。

例：「ます」

助動詞 特殊・ヌ 【活用形】

＃ 否定の助動詞「ぬ」の活用型。

例：「ぬ」

助動詞 不変化型 【活用形】

＃ 現代では活用しない助動詞。活用が想定できない口語や方言も含む。

例：「う」「よう」「まい」「(いざ行か)ん(む)」「(わから)ん<口語>」「(賜ラ)ン」「(美しい/学生)じゃ(ないか/ありません)<口語>」「~(美しい/学生)じゃん<口語>」「(~する)なかれ」「(使いたいん)や<「だ」の方言>」「やろ<方言>」「(いい)っす<口語>」「(いか)ねぇ<口語>」「(取れ)ねえ<口語>」「(負けてなら)じ」

助動詞 文語・?? 【活用形】

＃ 文語の助動詞。現在定義されている活用型は次の通り。「文語・ベシ」「文語・ゴトシ」「文語・ナリ」「文語・マジ」「文語・シム」「文語・キ」「文語・ケリ」「文語・ル」

例：「べし」「ごとし」「如し」「たり」「なり」「まじ」「き」「けり」「り」「る」

＊ IPA 品詞体系では、「じ」のための活用型が用意されているが、実際には不変化であるので、ここでは不変化型とした。

4.10 感動詞

感動詞

＃ 感動詞。あいさつなど。

例：「おはよう」「おはようございます」「こんにちは」「こんばんは」「ありがとう」「どうもありがとう」「ありがとうございます」「いただきます」「ごちそうさま」「さよなら」「さようなら」「はい」「いいえ」「ごめん」「ごめんなさい」...

4.11 記号

記号-一般

＃ 以下の分類以外の一般的な記号。

例：「○」「◎」「@」「$」「〒」「→」「+」など。

記号-アルファベット

＃ 英語のアルファベット。大文字、小文字。

例：「A」「a」

記号-句点

いわゆる句点。

例：「。」「 」

記号-読点

いわゆる読点。

例：「、」「 」

記号-空白

全角の空白文字 (画面上には見えない)。

記号-括弧開

例：「(」「{ 」「〔 」 「“ 」 「『 」「【 」...

記号-括弧閉

例：「)」「 }」「 ’ 」「 ”」 「 』」「 】」...

4.12 フィラー

フィラー

話し言葉で起こるあいずちや挿入的な音声ことば

例：「あの」「うんと」「えと」

4.13 その他

その他-間投

[名詞-接尾] や、[助詞-終助詞] としにくいもの。

例：「(だ) ァ」

参考文献

[1] 松本裕治, 北内啓, 山下達雄, 平野善隆, 松田寛, 浅原正幸. 日本語形態素解析システム『茶筌』version 2.0 使用説明書 第二版. Information Science Technical Report NAIST-IS-TR99012, Nara Institute of Science and Technology, 1999.

付 録C　ChaWanのメンテナンス法†

n_unit.c には、数詞と助数詞の変化が与えられている。このソースをメンテナンスするために、make_unit.c というプログラムがあり、unit_table.dat からこれを作成することができる。以下に標準添付の unit_table.dat の内容を示す。

```
% パーセント A2 -1 一({イチ/イッ}パーセント) 六({ロク/ロッ}パーセント)
ｃａｌ カロリー A4 -1
ｃｃ シーシー A2 -1
ｃｍ センチメートル A5 -1
ｄｂ デシベル A1 -1
ｄｋ ディーケー A1 -1
ｆｅｅｔ フィート A1 -1
ｇ グラム A1 -1
ｇｂ ギガバイト A1 -1
gバイト ギガバイト A1 -1
ｈｐａ ヘクトパスカル A1 -1
ｈｚ ヘルツ A1 -1
ｋｂ キロバイト A4 -1
ｋｇ キログラム A4 -1
ｋｈｚ キロヘルツ A4 -1
ｋｌ キロリットル A4 -1
ｋｍ キロメートル A4 -1
ｋｔ キロトン A4 -1
ｋｗ キロワット A1 -1
kグラム キログラム A4 -1
kバイト キロバイト A4 -1
kヘルツ キロヘルツ A4 -1
kメートル キロメートル A4 -1
```

† 天白 成一 ((株) アルカディア)

ｋリットル キロリットル A4 -1
ｋワット キロワット A4 -1
ｌ リットル A1 -1
ｍａ ミリアンペア A1 -1
ｍｂ メガバイト A1 -1
ｍｇ ミリグラム A1 -1
ｍｈｚ メガヘルツ A1 -1
ｍｍ ミリメートル A1 -1
ｍｏｌ モル A1 -1
ｍｔ メガトン A1 -1
ｍｖ ミリボルト A1 -1
ｍｗ ミリワット A1 -1
mバイト メガバイト A1 -1
mヘルツ メガヘルツ A1 -1
ｐａ パスカル A1 -1
ｐｐ ページ A2 -1 一（{イチ/イッ}ページ）六（{ロク/ロッ}ページ）八（{ハチ/ハッ}ページ）
ｐｐｍ ピーピーエム A1 -1
ｔ トン A3 -1 八（{ハチ/ハッ}トン）
ｖ ボルト A1 -1
ｗ ワット A1 -1
か月 カゲツ A6 -1
か国 カコク A6 -1
か所 カショ A6 -1
か条 カジョー A6 -1
か年 カネン A6 -1
くだり クダリ C1 10
けた ケタ A4 10 一（ヒトケタ）二（フタケタ）三（{ミ/サン}ケタ）四（{ヨ/ヨン}ケタ）五（{ゴ/イツ}ケタ）八（{ハチ/ハッ}ケタ）
すじ スジ C1 10
そろい ソロイ C3 10
つ ツ C1 10
つかみ ツカミ A2 2 一（ヒトツカミ）二（フタツカミ）

つまみ ツマミ C1 10
とおり トーリ C5 10 三({ミ/サン}トーリ) 四({ヨ/ヨン}トーリ) 五(ゴトーリ) 六(ロクトーリ) 八({ハチ/ハッ}トーリ)
ところ トコロ C2 10
とせ トセ C1 10
まわり マワリ C4 10 五({イツ/ゴ}マワリ) 十({ジュー/ト}マワリ)
アール アール A1 -1
アウト アウト E1 10
アンダー アンダー E1 10
アンペア アンペア A1 -1
イニング イニング A1 -1
インチ インチ A1 -1
エーカー エーカー A1 -1
オーバー オーバー E1 10
オーム オーム A1 -1
オクターブ オクターブ A1 -1
オンス オンス A1 -1
カラット カラット A2 -1
カロリー カロリー A4 -1
カ月 カゲツ A6 -1
カ国 カコク A6 -1
カ所 カショ A6 -1
カ条 カジョー A6 -1
カ年 カネン A6 -1
ガロン ガロン A1 -1
キュリー キュリー A1 -1
キロ キロ A4 -1
キログラム キログラム A4 -1
キロトン キロトン A4 -1
キロバイト キロバイト A4 -1
キロヘルツ キロヘルツ A4 -1
キロメートル キロメートル A4 -1
キロリットル キロリットル A4 -1

キロワット キロワット A4 -1
キロワット時 キロワットジ A4 -1
ギガバイト ギガバイト A1 -1
ギルダー ギルダー A1 -1
クラス クラス A2 2 一({ヒト/イチ}クラス) 二({フタ/ニ}クラス) 八({ハチ/ハッ}クラス)
グラム グラム A1 -1
グループ グループ A1 2 一({イチ/ヒト}グループ) 二({ニ/フタ}グループ)
ケタ ケタ A4 10 一(ヒトケタ) 二(フタケタ) 三({ミ/サン}ケタ) 四({ヨ/ヨン}ケタ) 五({ゴ/イツ}ケタ) 八({ハチ/ハッ}ケタ)
ケ月 カゲツ A6 -1
ケ国 カコク A6 -1
ケ所 カショ A6 -1
ケ条 カジョー A6 -1
ケ年 カネン A6 -1
サイクル サイクル A2 -1
シーシー シーシー A2 -1
シーズン シーズン A2 -1 一({イチ/イッ/ヒト}シーズン) 二({ニ/フタ}シーズン) 八({ハチ/ハッ}シーズン)
シート シート A2 -1
センチ センチ A5 -1
センチメートル センチメートル A5 -1 一({イチ/イッ}センチメートル)
セント セント A5 -1
タル タル A1 2 一(ヒトタル) 二(フタタル)
ダース ダース A1 -1
ダイン ダイン A1 -1
ディーケー ディーケー A1 -1
デシベル デシベル A1 -1
デニール デニール A1 -1
トライ トライ E1 10
トン トン A3 -1 八({ハチ/ハッ}トン)
ドル ドル A1 -1
ノット ノット A1 -1

バーツ バーツ A1 -1
バーディー バーディー E1 10
バール バール A1 -1
バッグ バッグ E1 10
バレル バレル A1 -1
パーセント パーセント A2 -1 一({イチ/イッ}パーセント) 六({ロク/ロッ}パーセント)
パスカル パスカル A1 -1
パット パット E1 10
ピーピーエム ピーピーエム A1 -1
フィート フィート A1 -1
フィート フィート E1 10
フォン フォン A1 -1
フラン フラン A1 -1 十({ジッ/ジュッ/ジュー}フラン)
フロア フロア E1 10
ブロック ブロック A1 -1
ヘクタール ヘクタール A1 -1
ヘクトパスカル ヘクトパスカル A1 -1
ヘルツ ヘルツ A1 -1
ページ ページ A2 -1 一({イチ/イッ}ページ) 六({ロク/ロッ}ページ) 八({ハチ/ハッ}ページ)
ペソ ペソ A2 -1
ペニー ペニー A1 -1
ホン ホン A2 -1
ボギー ボギー E1 10
ボルト ボルト A1 -1
ポイント ポイント A2 -1
ポンド ポンド A2 -1 一({イチ/イッ}ポンド)
マイル マイル A1 -1
マルク マルク A1 -1
ミクロン ミクロン A1 -1
ミリ ミリ A1 -1
ミリアンペア ミリアンペア A1 -1

ミリグラム ミリグラム A1 -1
ミリバール ミリバール A1 -1
ミリボルト ミリボルト A1 -1
ミリメートル ミリメートル A1 -1
ミリワット ミリワット A1 -1
メートル メートル A1 -1
メガ メガ A1 -1
メガサイクル メガサイクル A1 -1
メガトン メガトン A1 -1
メガバイト メガバイト A1 -1
メガヘルツ メガヘルツ A1 -1
モル モル A1 -1
ヤード ヤード A1 -1
ヤール ヤール A1 -1
ラン ラン E1 10
リッター リッター A1 -1
リットル リットル A1 -1
ルーブル ルーブル A1 -1
ルクス ルクス A1 -1
ルピー ルピー A1 -1
ワット ワット A1 -1
ヵ月 カゲツ A6 -1
ヵ国 カコク A6 -1
ヵ所 カショ A6 -1
ヵ条 カジョー A6 -1
ヵ年 カネン A6 -1
ヶ月 カゲツ A6 -1
ヶ国 カコク A6 -1
ヶ所 カショ A6 -1
ヶ条 カジョー A6 -1
ヶ年 カネン A6 -1
Ω オーム A1 -1
握り ニギリ A1 2 一（ヒトニギリ）二（フタニギリ）

位 イ A1 3 三(サン{イ/ミ})
羽 バ A1 -1 一(イチワ) 二(ニワ) 三(サン{ワ/バ}) 四(ヨン{ワ/バ}) 五(ゴワ) 六({ロッパ/ロクワ}) 七({ナナ/シチ}ワ) 八({ハチワ/ハッパ}) 九(キューワ) 十({ジューワ/ジュッパ/ジッパ}) 百(ヒャッパ) 三百(ビャッパ)
羽 ワ A4 -1 三(サン{ワ/バ}) 四(ヨン{ワ/バ}) 六({ロッパ/ロクワ}) 八({ハチワ/ハッパ}) 十({ジューワ/ジュッパ/ジッパ}) 百(ヒャッパ) 三百(ビャッパ) 千(センバ) 三千(ゼンバ) 万(マンバ) 何(ナンバ)
駅 エキ C4 10
円 エン B6 -1
億 オク A1 -1
価 カ A6 -1
家族 カゾク A4 -1 一({イチ/イッ/ヒト}カゾク) 二({ニ/フタ}カゾク)
科目 カモク A6 -1
箇月 カゲツ A6 -1
箇国 カコク A6 -1
箇所 カショ A6 -1
箇条 カジョー A6 -1
箇年 カネン A6 -1
課 カ A6 -1
画 カク A6 -1
回 カイ A6 -1 八({ハチ/ハッ}カイ)
回忌 カイキ A4 -1
回生 カイセー A6 -1
回戦 カイセン A6 -1
回転 カイテン A6 -1
海里 カイリ A4 -1
階 カイ A6 -1 八({ハチ/ハッ}カイ)
階級 カイキュー A4 -1 一({イチ/イッ}カイキュー) 六({ロッ/ロク}カイキュー)
階層 カイソー A4 -1
浬 カイリ A4 -1
角 カク A6 -1
学期 ガッキ A1 -1
学級 ガッキュー A1 -1

学区 ガック A1 -1
学年 ガクネン A1 -1
楽章 ガクショー A1 -1
割 ワリ A1 -1
株 カブ A4 3 一({イチ/ヒト}カブ) 二({ニ/フタ}カブ) 三({サン/ミ}カブ) 四({ヨン/ヨ}カブ) 五({ゴ/イツ}カブ)
巻 カン A6 -1
竿 サオ C2 10
管区 カンク A6 -1
缶 カン A4 2 一({イチ/ヒト}カン) 二({ニ/フタ}カン)
貫 カン A6 -1
貫目 カンメ A6 -1
間 ケン A6 -1
基 キ A6 -1 八({ハチ/ハッ}キ)
期 キ A6 -1 八({ハチ/ハッ}キ)
期生 キセー A6 -1
機 キ A6 -1 八({ハチ/ハッ}キ)
機種 キシュ A4 -1
騎 キ A6 -1 八({ハチ/ハッ}キ)
客 キャク A6 -1 八({ハチ/ハッ}キャク)
脚 キャク A6 -1 八({ハチ/ハッ}キャク)
球 キュー A6 -1 八({ハチ/ハッ}キュー)
球団 キューダン A2 -1
球目 キューメ A6 -1 八({ハチ/ハッ}キューメ)
級 キュー A6 -1 八({ハチ/ハッ}キュー)
強 キョー A6 -1
橋 ハシ A2 -1
局 キョク A6 -1 八({ハチ/ハッ}キョク)
曲 キョク A6 -1 八({ハチ/ハッ}キョク)
極 キョク A6 -1 八({ハチ/ハッ}キョク)
玉 タマ C1 10 三({ミ/サン}タマ) 四({ヨ/ヨン}タマ) 八({ハチ/ハッ}タマ)
粁 キロメートル A4 -1
斤 キン A6 -1 三(サン{キン/ギン})

筋 スジ C1 10
句 ク A6 -1 八({ハチ/ハッ}ク)
区間 クカン A4 -1 一({イチ/ヒト/イッ}クカン) 二({ニ/フタ}クカン) 八({ハチ/ハッ}クカン)
型 ガタ A1 -1
系統 ケイトー A4 -1 一({イチ/イッ}ケイトー) 六({ロッ/ロク}ケイトー)
桁 ケタ A4 10 一(ヒトケタ) 二(フタケタ) 三({ミ/サン}ケタ) 四({ヨ/ヨン}ケタ) 五({ゴ/イツ}ケタ) 八({ハチ/ハッ}ケタ)
月 ガツ B1 -1
月号 ガツゴー B1 -1
月末 ガツマツ B1 -1
件 ケン A6 -1
検体 ケンタイ A4 -1
県 ケン A6 -1
軒 ケン A6 -1
元 ゲン B6 -1
弦 ゲン B3 -1
言 コト C4 10
個 コ A6 -1 八({ハチ/ハッ}コ)
個月 カゲツ A6 -1
個口 コグチ A6 -1
個国 カコク A6 -1
個所 カショ A6 -1
個条 カジョー A6 -1
個年 カネン A6 -1
戸 コ A6 -1 八({ハチ/ハッ}コ)
語 ゴ A1 -1
光年 コーネン A4 -1 一({イチ/イッ}コーネン) 六({ロク/ロッ}コーネン)
口 クチ A4 3 一(ヒトクチ) 二(フタクチ) 三(ミクチ) 四({ヨ/ヨン}クチ)
口径 コーケー A4 -1 一({イチ/イッ}コーケー) 六({ロク/ロッ}コーケー)
工程 コーテー A4 -1 一({イチ/イッ}コーテー) 六({ロク/ロッ}コーテー)
校 コー A6 -1
行 {コー/ギョー} A1 -1 一({イッコー/イチギョウ}) 六({ロッコー/ロクギョー

}）八（{ハッコー/ハチコー/ハチギョー}）十（{ジュッコー/ジッコー/ジューギョー}）

行 ギョー A1 -1

行 コー A7 -1

項 コー A6 -1

項目 コーモク A4 -1 一（{イチ/イッ}コーモク）六（{ロク/ロッ}コーモク）

号 ゴー A1 -1

号館 ゴーカン A1 -1

号機 ゴーキ A1 -1

号室 ゴーシツ A1 -1

号証 ゴーショー A1 -1

号線 ゴーセン A1 -1

号棟 ゴートー A1 -1

合 ゴー A1 -1

差し サシ C2 10

才 サイ A5 -1

才時 サイジ A5 -1

歳 サイ A5 -1

歳児 サイジ A5 -1

歳時 サイジ A5 -1

冊 サツ A5 -1

刷 サツ A5 -1

皿 サラ A2 10 四（{ヨ/ヨン}サラ）五（{イツ/ゴ}サラ）六（{ロク/ム}サラ）十（{ジッ/ジュッ/ト}サラ）

山 ヤマ C1 10

酸化 サンカ A2 -1 一（{イチ/イッ}サンカ）

市 シ A5 -1

紙 シ A5 -1

試合 シアイ A2 2 一（{イッ/ヒト}シアイ）二（{ニ/フタ}シアイ）八（{ハッ/ハチ}シアイ）

誌 シ A5 -1

児 ジ A1 -1 四（{ヨ/ヨン}ジ）

字 ジ B4 -1

時 ジ B4 -1 九（クジ）
時すぎ ジスギ B4 -1 九（クジスギ）
時過ぎ ジスギ B4 -1 九（クジスギ）
時間 ジカン B4 -1 九（クジカン）
時限 ジゲン B4 -1 九（クジゲン）
時半 ジハン B4 -1
次 ジ B4 -1 九（クジ）
次元 ジゲン B4 -1 九（クジゲン）
式 シキ A5 -1
室 シツ A5 -1
社 シャ A5 -1
車線 シャセン A5 -1
勺 シャク A5 -1 三（サンジャク）
尺 シャク A5 -1 三（サンジャク）
手 テ A5 -1
種 シュ A5 -1
種目 シュモク A5 -1
種類 シュルイ A5 2 一（{イッ/ヒト}シュルイ）二（{ニ/フタ}シュルイ）
首 シュ A5 -1
周 シュー A5 -1
周忌 シューキ A5 -1
周期 シューキ A5 -1
周年 シューネン A5 -1
周目 シューメ A5 -1
州 シュー A5 -1
週 シュー A5 -1
週間 シューカン A5 -1
集 シュー A5 -1
重 ジュー B5 -1
巡 ジュン A1 -1
女 ジョ B3 2 二（{ニ/ジ}ジョ）
勝 ショー A5 -1
升 ショー A5 -1

小節 ショーセツ A5 -1
床 ショー A5 -1
省 ショー A5 -1
章 ショー A5 -1
乗 ジョー A1 -1 四({ヨン/ヨ}ジョー)
場所 バショ C6 10
条 ジョー A1 -1 四({ヨン/シ}ジョー)
畳 ジョー A1 -1 四({ヨン/ヨ}ジョー) 九({キュー/ク}ジョー)
錠 ジョー A1 -1 四({ヨン/ヨ}ジョー)
色 イロ C1 10 四({ヨ/ヨン}イロ) 五({ゴ/イツ}イロ) 六({ム/ロク}イロ) 八({ヤ/ハチ}イロ) 九({ココノ/キュー}イロ) 十({ト/ジュー}イロ)
色 ショク A5 -1
食 ショク A5 -1
審 シン A5 -1
親等 シントー A5 -1
進 シン A5 -1
進数 シンスー A5 -1
針 ハリ C2 10
人 ニン B4 2 一({ヒトリ/イチニン}) 二({フタリ/ニニン})
人月 ニンゲツ B4 -1
人称 ニンショー B4 -1
人前 ニンマエ B4 -1
人組 ニングミ B4 -1
陣 ジン B6 -1
寸 スン A5 -1 三(サン{スン/ズン})
世 セー A5 -1
世紀 セーキ A5 -1
世紀 セイキ A5 -1
世帯 セタイ A5 -1
世代 セダイ A5 -1
隻 セキ A5 -1
石 {コク/セキ} A5 -1
石 コク A5 -1

石 セキ A5 -1
切れ キレ C2 10 二 ({フタ/ニ}キレ) 四 ({ヨン/ヨ}キレ) 五 ({ゴ/イツ}キレ) 六 ({ロッ/ム}キレ) 八 ({ハチ/ヤ}キレ) 十 ({ジッ/ジュッ/ト}キレ)
節 セツ A5 -1
戦 セン A5 -1
選 セン A5 -1
選手 センシュ A5 -1
銭 セン A5 -1
膳 ゼン A1 -1
糎 センチメートル A5 -1
組 クミ A4 10 一 ({イチ/ヒト}クミ) 二 ({ニ/フタ}クミ) 三 ({ミ/サン}クミ) 四 ({ヨン/ヨ}クミ) 五 ({ゴ/イツ}クミ) 十 ({ジッ/ジュッ/ト}クミ)
層 ソー A5 -1
束 タバ C3 10 四 ({ヨ/ヨン}タバ) 五 ({イツ/ゴ}タバ) 十 ({ジッ/ジュッ/ト}タバ)
足 ソク A5 -1
続き ツヅキ A2 2 一 (ヒトツヅキ) 二 (フタツヅキ)
揃 ソロイ C3 10
打 ダ A1 -1
打席 ダセキ A1 -1
打点 ダテン A1 -1
体 タイ A5 -1
対 タイ A5 -1
袋 フクロ C4 10
代 ダイ A1 -1
代目 ダイメ A1 -1
台 ダイ A1 -1 四 ({ヨン/ヨ}ダイ) 九 ({キュー/ク}ダイ)
大 ダイ A1 -1
題 ダイ A1 -1
卓 タク A2 -1
樽 タル A1 2 一 (ヒトタル) 二 (フタタル)
段 ダン B4 -1 四 ({ヨン/ヨ}ダン) 九 ({キュー/ク}ダン)
段位 ダンイ B4 -1
段階 ダンカイ A1 -1 四 ({ヨン/ヨ}ダンカイ)

段式 ダンシキ A1 -1 四({ヨン/ヨ}ダンシキ)
男 ナン B3 2 二({ニ/ジ}ナン)
地区 チク A2 -1 一({イチ/イッ}チク) 八({ハチ/ハッ}チク)
地点 チテン A2 -1 八({ハチ/ハッ}チテン)
着 チャク A5 -1
柱 ハシラ C2 10 五({イツ/ゴ}ハシラ) 六({ム/ロク}ハシラ) 八({ヤ/ハチ}ハシラ)
丁 チョー A5 -1
丁目 チョーメ A5 -1
帖 ジョー A1 -1 四({ヨン/ヨ}ジョー)
町 チョー A5 -1
通 ツー A5 -1
通り トーリ C5 10 三({ミ/サン}トーリ) 四({ヨ/ヨン}トーリ) 五(ゴトーリ) 六(ロクトーリ) 八({ハチ/ハッ}トーリ)
掴み ツカミ A2 2 一(ヒトツカミ) 二(フタツカミ)
坪 ツボ C1 10 三({サン/ミ}ツボ) 四({ヨン/ヨ}ツボ) 八({ハチ/ハッ}ツボ) 十({ジッ/ジュッ/ト}ツボ)
滴 テキ A5 -1
店 テン A5 -1
店舗 テンポ A5 -1
点 テン A5 -1
斗 ト A1 -1
都 ト A1 -1
度 {タビ/ド} C1 10 一({ヒトタビ/イチド}) 二({フタタビ/ニド}) 三({ミタビ/サンド}) 四({ヨタビ/ヨンド}) 五({イツタビ/ゴタビ/ゴド}) 六({ムタビ/ロクド}) 七({ナナタビ/ナナド/シチド}) 八({ヤタビ/ハチド}) 九({キュータビ/クタビ/キュード/クド}) 十({ジュッタビ/ジッタビ/トタビ/ジュード})
度 タビ C1 10 五({イツ/ゴ}タビ) 九({キュー/ク}タビ) 十({ジッ/ジュッ/ト}タビ)
度 ド A1 -1 四({ヨン/ヨ}ド) 九({キュー/ク}ド)
度目 ドメ A1 -1 四({ヨン/ヨ}ドメ) 九({キュー/ク}ドメ)
党 トー A5 -1
棟 {トー/ムネ} C1 10 一({イチトー/イットー/ヒトムネ}) 二({ニトー/フタムネ

}）三（{サントー/サンムネ/ミムネ}）四（{ヨントー/ヨンムネ/ヨムネ}）五（{ゴトー/ゴムネ/イツムネ}）六（{ロクトー/ロクムネ}）七（{ナナトー/シチトー/ナナムネ/シチムネ}）八（{ハチトー/ハットー/ハチムネ}）九（キュー{トー/ムネ}）十（{ジュットー/ジットー/ジュームネ/トムネ}）

棟 トー A5 -1

棟 ムネ C1 10 三（{サン/ミ}ムネ）四（{ヨン/ヨ}ムネ）五（{ゴ/イツ}ムネ）十（{ジュー/ト}ムネ）

灯 トー A5 -1

等 トー A5 -1

等級 トーキュー A3 -1 八（{ハッ/ハチ}トーキュー）

等親 トーシン A3 -1 八（{ハッ/ハチ}トーシン）

等席 トーセキ A3 -1 八（{ハッ/ハチ}トーセキ）

等地 トーチ A3 -1 八（{ハッ/ハチ}トーチ）

等分 トーブン A3 -1 八（{ハッ/ハチ}トーブン）

等辺 トーヘン A3 -1 八（{ハッ/ハチ}トーヘン）

頭 トー A5 -1

頭身 トーシン A3 -1 八（{ハッ/ハチ}トーシン）

噸 トン A3 -1 八（{ハチ/ハッ}トン）

日 カ A1 10 一（{ツイタチ/イチニチ}）二（{フツカ/ニニチ}）三（{ミッカ/サンニチ}）四（{ヨッカ/ヨンニチ}）五（{イツカ/ゴニチ}）六（{ムイカ/ロクニチ}）七（{ナノカ/ナヌカ/ナナニチ}）八（{ヨーカ/ハチニチ}）九（{ココノカ/キューニチ/クニチ}）十（{トーカ/ジューニチ}）

日 ジツ A1 10 一（{ツイタチ/イチニチ}）二（{フツカ/ニニチ}）三（{ミッカ/サンニチ}）四（{ヨッカ/ヨンニチ}）五（{イツカ/ゴニチ}）六（{ムイカ/ロクニチ}）七（{ナノカ/ナヌカ/ナナニチ}）八（{ヨーカ/ハチニチ}）九（{ココノカ/キューニチ/クニチ}）十（{トーカ/ジューニチ}）

日 チ A1 10 一（{ツイタチ/イチニチ}）二（{フツカ/ニニチ}）三（{ミッカ/サンニチ}）四（{ヨッカ/ヨンニチ}）五（{イツカ/ゴニチ}）六（{ムイカ/ロクニチ}）七（{ナノカ/ナヌカ/ナナニチ}）八（{ヨーカ/ハチニチ}）九（{ココノカ/キューニチ/クニチ}）十（{トーカ/ジューニチ}）

日 ニチ A1 10 一（{ツイタチ/イチニチ}）二（{フツカ/ニニチ}）三（{ミッカ/サンニチ}）四（{ヨッカ/ヨンニチ}）五（{イツカ/ゴニチ}）六（{ムイカ/ロクニチ}）七（{ナノカ/ナヌカ/ナナニチ}）八（{ヨーカ/ハチニチ}）九（{ココノカ/キューニチ/

クニチ}）十（{トーカ/ジューニチ}）
日ぶり ニチブリ B5 -1
日号 ニチゴー B5 -1
日付 ニチヅケ B5 -1
日目 ニチメ B5 -1
年 ネン B4 -1
年ぶり ネンブリ B4 -1
年間 ネンカン B4 -1
年後 ネンゴ B4 -1
年生 ネンセー B4 -1
年代 ネンダイ B4 -1
年度 ネンド B4 -1
年余 {ネンアマリ/ネンヨ} B4 -1
把 ワ A2 -1 三（サン{ワ/バ}）四（ヨン{ワ/バ}）六（{ロッパ/ロクワ}）八（{ハチワ/ハッパ}）十（{ジューワ/ジュッパ/ジッパ}）百（ヒャッパ）三百（ビャッパ）千（センバ）三千（ゼンバ）万（マンバ）何（ナンワ）
波 ハ A6 -1 一（イッパ）六（{ロッパ/ロクハ}）八（{ハッパ/ハチハ}）十（{ジュッ/ジッ}パ）
馬身 バシン A1 -1
馬力 バリキ A1 -1
敗 ハイ A7 -1 一（イッパイ）六（{ロッパイ/ロクハイ}）八（{ハッパイ/ハチハイ}）十（{ジュッ/ジッ}パイ）
杯 ハイ A7 -1 一（イッパイ）六（{ロッパイ/ロクハイ}）八（{ハッパイ/ハチハイ}）十（{ジュッ/ジッ}パイ）
倍 バイ A1 -1
拍 ハク A6 -1 一（{イッパク/ヒトハク}）二（{ニ/フタ}ハク）三（サンパク）四（ヨン{ハク/パク}）六（{ロッパク/ロクハク}）八（{ハッパク/ハチハク}）十（{ジュッ/ジッ}パク）百（ヒャッパク）三百（ビャッパク）何（ナンパク）
拍子 ビョーシ A1 -1
泊 ハク A6 -1 一（{イッパク/ヒトハク}）二（{ニ/フタ}ハク）三（サンパク）四（ヨン{ハク/パク}）六（{ロッパク/ロクハク}）八（{ハッパク/ハチハク}）十（{ジュッ/ジッ}パク）百（ヒャッパク）三百（ビャッパク）何（ナンパク）
箱 ハコ C3 10 三（{ミハコ/サンバコ}）四（{ヨ/ヨン}ハコ）五（{ゴ/イツ}ハコ）六

({ロクハコ/ロッパコ}) 八 ({ハチハコ/ハッパコ}) 十 ({ジッパコ/ジュッパコ/トハコ}) 百 (ヒャッパコ) 三百 (ビャッパコ) 千 (センバコ) 三千 (ゼンバコ) 万 (マンバコ)

鉢 ハチ C2 10 四 ({ヨ/ヨン}ハチ) 十 ({ジッパチ/ジュッパチ/トハチ})

発 ハツ A7 -1 一 (イッパツ) 三 (サンパツ) 四 (ヨン{パツ/ハツ}) {ロッパツ/ロクハツ} 十 ({ジュッ/ジッ}パツ) 百 (ヒャッパツ) 三百 (ビャッパツ) 千 (センパツ) 三千 (ゼンパツ) 何 (ナンパツ)

反 タン A5 -1

版 ハン A6 -1 六 ({ロッパン/ロクハン}) 八 ({ハッパン/ハチハン}) 十 ({ジュッ/ジッ}パン) 百 (ヒャッパン) 三百 (ビャッパン)

犯 ハン A6 -1 六 ({ロッパン/ロクハン}) 八 ({ハッパン/ハチハン}) 十 ({ジュッ/ジッ}パン) 百 (ヒャッパン) 三百 (ビャッパン)

班 ハン A6 -1 六 ({ロッパン/ロクハン}) 八 ({ハッパン/ハチハン}) 十 ({ジュッ/ジッ}パン) 百 (ヒャッパン) 三百 (ビャッパン)

晩 バン C1 10

番 バン A1 -1 四 ({ヨン/ヨ}バン) 七 ({ナナ/シチ}バン) 九 ({キュー/ク}バン)

番手 バンテ B6 -1 四 ({ヨン/ヨ}バンテ)

番線 バンセン A1 -1

番台 バンダイ A1 -1

番地 バンチ A1 -1

番目 バンメ A1 -1 四 ({ヨン/ヨ}バンメ) 九 ({キュー/ク}バンメ)

尾 ビ A1 -1

匹 ヒキ A6 -1 一 (イッピキ) 三 (サンビキ) 四 ({ヨン/シ}ヒキ) 六 ({ロッピキ/ロクヒキ}) 八 ({ハチヒキ/ハッピキ}) 十 ({ジュッ/ジッ}ピキ) 百 (ヒャッピキ) 三百 (ビャッピキ) 千 (センビキ) 三千 (ゼンビキ) 万 (マンビキ) 何 (ナンビキ)

俵 ヒョー A6 -1 一 (イッピョー) 三 (サンビョー) 六 ({ロッピョー/ロクヒョー}) 八 ({ハチヒョー/ハッピョー}) 十 ({ジュッ/ジッ}ピョー) 百 (ヒャッピョー) 三百 (ビャッピョー) 千 (センビョー) 三千 (ゼンビョー) 万 (マンビョー) 何 (ナンビョー)

票 ヒョー A6 -1 一 (イッピョー) 三 (サンビョー) 六 ({ロッピョー/ロクヒョー}) 八 ({ハチヒョー/ハッピョー}) 十 ({ジュッ/ジッ}ピョー) 百 (ヒャッピョー) 三百 (ビャッピョー) 千 (センビョー) 三千 (ゼンビョー) 万 (マンビョー) 何 (ナンビョー)

秒 ビョー A1 -1

品 {ヒン/シナ} A7 10 一 ({イッピン/ヒトシナ}) 二 ({ニヒン/フタシナ}) 三 ({サ

ンピン/ミシナ}）六（{ロッピン/ロクヒン}）八（{ハッピン/ハチヒン}）十（{ジュッピン/ジッピン/トシナ}）何（ナン{ヒン/ピン}）

品 シナ A2 10 一（ヒトシナ）二（フタシナ）三（{ミ/サン}シナ）四（{ヨン/ヨ}シナ）

品 ヒン A7 -1 一（イッピン）三（サンピン）六（{ロッピン/ロクヒン}）八（{ハッピン/ハチヒン}）何（ナン{ヒン/ピン}）

品種 ヒンシュ A1 -1 十（{ジュー/ジッ/ジュッ}ヒンシュ）

品目 ヒンモク A1 -1 十（{ジュー/ジッ/ジュッ}ヒンモク）

部 ブ A1 -1

部屋 ヘヤ C4 10 四（{ヨン/ヨ}ヘヤ）五（{ゴ/イツ}ヘヤ）

部制 ブセー A1 -1

分 {ブ/ブン/フン} A1 -1 一（{イチブ/イチブン/イップン}）三（{サンブ/サンブン/サンプン}）六（{ロクブ/ロクブン/ロップン/ロクフン}）八（{ハチブ/ハチブン/ハチフン/ハップン}）九（{キューブ/クブ/キューブン/キューフン}）十（{ジューブ/ジューブン/ジュップン/ジップン}）百（{ヒャクブ/ヒャクブン/ヒャップン}）三百（{ビャクブ/ビャクブン/ビャップン}）何（ナン{ブ/ブン/プン}）

分 フン A6 -1 一（イップン）三（サンプン）六（{ロップン/ロクフン}）八（{ハチフン/ハップン}）十（{ジュッ/ジッ}プン）百（ヒャップン）三百（ビャップン）何（ナンプン）

分 ブ A1 -1 九（{キュー/ク}ブ）

分木 ブンギ A1 -1

分目 フンメ A6 -1 一（イップンメ）三（サンプンメ）六（{ロップンメ/ロクフンメ}）八（{ハチフンメ/ハップンメ}）十（{ジュッ/ジッ}プンメ）百（ヒャップンメ）三百（ビャップンメ）何（ナンプンメ）

分野 ブンヤ A1 -1

文 ブン A1 -1

文 モン A1 -1

平米 ヘーベー A1 -1

平方 ヘーホー A2 -1

平方センチメートル ヘーホーセンチメートル A1 -1

平方メートル ヘーホーメートル A1 -1

片 {ヘン/ペン} A6 -1 一（イッペン）三（サン{ペン/ベン}）六（{ロッペン/ロクヘン}）八（{ハチヘン/ハッペン}）十（{ジュッ/ジッ}ペン）百（ヒャッペン）三百（ビャッ

ペン）千（センペン）何（ナンペン）
片 ヘン A6 -1 一（イッペン）三（サン{ペン/ベン}）六（{ロッペン/ロクヘン}）八（{ハチヘン/ハッペン}）十（{ジュッ/ジッ}ペン）百（ヒャッペン）三百（ビャッペン）千（センペン）何（ナンペン）
篇 ヘン A6 -1 一（イッペン）三（サン{ペン/ベン}）六（{ロッペン/ロクヘン}）八（{ハチヘン/ハッペン}）十（{ジュッ/ジッ}ペン）百（ヒャッペン）三百（ビャッペン）千（センペン）何（ナンペン）
編 ヘン A6 -1 一（イッペン）三（サン{ペン/ベン}）六（{ロッペン/ロクヘン}）八（{ハチヘン/ハッペン}）十（{ジュッ/ジッ}ペン）百（ヒャッペン）三百（ビャッペン）千（センペン）何（ナンペン）
辺 ヘン A6 -1 一（イッペン）三（サン{ペン/ベン}）六（{ロッペン/ロクヘン}）八（{ハチヘン/ハッペン}）十（{ジュッ/ジッ}ペン）百（ヒャッペン）三百（ビャッペン）千（センペン）何（ナンペン）
便 ビン A1 -1
歩 ホ A7 -1 一（イッポ）三（サンポ）六（{ロッポ/ロクホ}）八（{ハチホ/ハッポ}）十（{ジュッ/ジッ}ポ）百（ヒャッポ）三百（ビャッポ）千（センポ）三千（ゼンポ）何（ナンポ）
報 ホー A7 -1 一（イッポー）三（サンポー）六（{ロッポー/ロクホー}）八（{ハチホー/ハッポー}）十（{ジュッ/ジッ}ポー）
方 {ポー/ホー} A7 -1 一（イッポー）三（サンポー）六（{ロッポー/ロクホー}）八（{ハチホー/ハッポー}）十（{ジュッ/ジッ}ポー）
方 ホー A7 -1 一（イッポー）三（サンポー）六（{ロッポー/ロクホー}）八（{ハチホー/ハッポー}）十（{ジュッ/ジッ}ポー）
法 {ポー/ホー} A7 -1 一（イッポー）三（サンポー）六（{ロッポー/ロクホー}）八（{ハチホー/ハッポー}）十（{ジュッ/ジッ}ポー）
法 ホー A7 -1 一（イッポー）三（サンポー）六（{ロッポー/ロクホー}）八（{ハチホー/ハッポー}）十（{ジュッ/ジッ}ポー）
本 ホン A6 -1 一（イッポン）三（サンボン）四（{ヨン/シ}ホン）六（{ロッポン/ロクホン}）八（{ハチホン/ハッポン}）十（{ジュッ/ジッ}ポン）百（ヒャッポン）三百（ビャッポン）千（センボン）三千（ゼンボン）万（マンボン）何（ナンボン）
本立て ホンダテ A6 -1 一（イッポンダテ）三（サンボンダテ）四（{ヨン/シ}ホンダテ）六（{ロッポンダテ/ロクホンダテ}）八（{ハチホンダテ/ハッポンダテ}）十（{ジュッ/ジッ}ポンダテ）百（ヒャッポンダテ）三百（ビャッポンダテ）千（センボンダ

テ）三千（ゼンボンダテ）万（マンボンダテ）何（ナンボンダテ）
本塁打 ホンルイダ A2 -1
枚 マイ A1 -1 四（{ヨン/ヨ}マイ）
哩 マイル A1 -1
幕 マク C6 10 三（{サン/ミ}マク）四（{ヨン/ヨ}マク）七（{ナナ/シチ}マク）
幕目 マクメ C1 10 三（{サン/ミ}マクメ）四（{ヨン/ヨ}マクメ）七（{ナナ/シチ}マクメ）
粍 ミリメートル A1 -1
名 メー A1 -1 四（{ヨン/ヨ}メー）
名分 メーブン A1 -1 四（{ヨン/ヨ}メーブン）
名様 メーサマ A1 -1 四（{ヨン/ヨ}メーサマ）
面 メン A1 -1
毛 モー A1 -1
目 モク A1 -1
問 モン A1 -1
匁 モンメ A1 -1
夜 {ヤ/ヨ} B3 2 一（{ヒトヨ/イチヤ}）二（{フタヨ/ニヤ}）
夜 ヤ B3 -1
役 ヤク C6 10
里 リ B4 -1
立方 リッポー A1 -1
立方センチメートル リッポーセンチメートル A1 -1
立方メートル リッポーメートル A1 -1
粒 ツブ C1 10 三（{サン/ミ}ツブ）四（{ヨン/ヨ}ツブ）五（{ゴ/イツ}ツブ）八（{ハチ/ハッ}ツブ）十（{ジッ/ジュッ/ト}ツブ）
両 リョー A1 -1
厘 リン A1 -1
輪 リン A1 -1 四（{ヨン/ヨ}リン）
塁 ルイ A1 -1
類 ルイ A1 -1
列 レツ A1 -1
連 レン A1 -1
連休 レンキュー A1 -1

連結 レンケツ A1 -1
連載 レンサイ A1 -1
連勝 レンショー A1 -1
連戦 レンセン A1 -1
連続 レンゾク A1 -1
連打 レンダ A1 -1
連隊 レンタイ A1 -1
連投 レントー A1 -1
連覇 レンパ A1 -1
連破 レンパ A1 -1
連敗 レンパイ A1 -1
話 ワ A1 -1
棹 サオ C2 10
艘 ソー A5 -1