品詞タグ付きコーパスのための品詞体系変換ツール: Conbu

橋本泰一　野呂智哉　徳永健伸　田中穂積
東京工業大学 大学院情報理工学研究科 計算工学専攻
{taiichi, noro, take, tanaka}@cl.cs.titech.ac.jp

1 はじめに

近年，様々な言語資源 (コーパス) が利用可能になってきたこと，計算機の著しい性能向上などの理由から，自然言語処理技術は，確率・統計的な手法が主流になっている．品詞タグ付きコーパス，依存構造付きコーパス，構文木付きコーパスなどは，確率・統計手法のモデルの学習データとして用いられるため，自然言語処理では，非常に重要なものとなっている．

現在，様々な言語知識が付与されたコーパスが利用可能になっている．特に，形態素区切りと品詞は基本的な言語知識であるために，言語知識が付与されたコーパスであれば，必ず付与されている．しかし，日本語の場合，英語などと異なり形態素の区切りが明白でないため，コーパス毎に異なる品詞体系に基づき，形態素区切り，品詞タグ付けが行われ，コーパス間の品詞体系の互換性はないのが現状である．この問題を解決するために，品詞体系を変換する手法についての研究が行われるようになった．[2,3]

本論文では，品詞体系の変換規則を体系化し，その体系に基づき開発した品詞体系変換ツール Conbu について述べる．そして，このツールを用い，京都大学テキストコーパス [4]，RWC コーパス (修正版)[1] [1]，東工大コーパス EDR 版 [7] の 3 つの品詞体系を相互に変換する 4 種類の変換規則を作成した．さらに，作成した 4 種類の変換規則のうち，3 種類の変換規則について評価実験行い，その実験結果について述べる．

2 品詞体系変換ツール: Conbu

Conbu は，品詞タグ付けされたコーパスを入力とし，他の品詞体系へ変換するツールである．京都大学テキストコーパスの品詞体系に基づき品詞付けされた文を RWC コーパスの品詞体系に変換した例を図 1 に示す．Conbu は，品詞体系の変換規則とコーパスに対して変換規則を適用するモジュールの二つにより構成されている．変換規則は，人手により作成する．

[1]RWC コーパスの形態素区切りや品詞の間違いを人手により修正したコーパス

2.1 変換規則

2.1.1 基本形

変換規則は，形態素，読み，品詞の三つの情報を用いる．以後，本論文では，形態素，読み，品詞の三組は，括弧”()” を用いて，(” 形態素”，” 読み”，” 品詞”) と表現する．変換規則の記述方法は，次のように定義する．

$$P < M >< R > \Rightarrow\ P' < M' >< R' >$$

P, M, R は変換前の品詞，形態素，読みを，P', M', R' は変換後の品詞，形態素，読みを表す．例えば，(” 太郎”，” タロウ”，” 人名”) を (” 太郎”，” タロウ”，” 名詞-固有名詞-人名-名”) へ変換する規則は，例 1 のように記述する．

[例 1]
人名<太郎><タロウ> => 名詞-固有名詞-人名-名<太郎><タロウ>

2.1.2 結合・分割

変換規則の両辺には，連続する三組を条件として記述することができる．連続する三組は，カンマ”,” で区切って定義する．

$$P_1 < M_1 >< R_1 >, \ldots, P_m < M_m >< R_m >$$
$$\Rightarrow\quad P'_1 < M'_1 >< R'_1 >, \ldots, P'_n < M'_n >< R'_n >$$

連続する三組を条件として指定することにより，形態素の結合や分割が可能になる．例えば，連続する (” と”，” ト”，” 格助詞”)，(” いう”，” イウ”，” 動詞/子音動詞ワ行/基本形”) を結合して，(” という”，” トイウ”，” 助詞-格助詞-連語”) へ変換したい場合は，例 2 のように記述する．

[例 2]
格助詞<と><ト>,
動詞/子音動詞ワ行/基本形<いう><イウ>
=> 助詞-格助詞-連語<という><トイウ>

また，(” 買って”，” カッテ”，” 動詞/子音動詞ワ行/タ系連用テ形”) を分割して，二つの三組 (” 買っ”，” カッ”，” 動詞-自立/五段・ワ行促音便/連用タ接続”)，(” て”，” テ”，” 助詞-接続助詞”) に変換したい場合は，例 3 のように記述する．

変換前
品詞体系：京都大学テキストコーパス

形態素	読み	品詞
太郎	タロウ	人名
は	ハ	副助詞
、	、	読点
三国志	サンゴクシ	固有名詞
と	ト	格助詞
いう	イウ	動詞/子音動詞ワ行/基本形
本	ホン	普通名詞
を	ヲ	格助詞
買って	カッテ	動詞/子音動詞ワ行/タ系連用テ形
から	カラ	接続助詞
１００	イチゼロゼロ	数詞
回	カイ	名詞性名詞助数
以上	イジョウ	名詞性名詞接尾
読んで	ヨンデ	動詞/子音動詞マ行/タ系連用テ形
いる	イル	動詞性接尾辞/母音動詞/基本形
。	。	句点

EOS

変換後
品詞体系：RWCコーパス

形態素	読み	品詞
太郎	タロウ	名詞-固有名詞-人名-名
は	ハ	助詞-係助詞
、	、	記号-読点
三国志	サンゴクシ	名詞-固有名詞-一般
という	トイウ	助詞-格助詞-連語
本	ホン	名詞-一般
を	ヲ	名詞-格助詞-一般
買っ	カッ	動詞-自立/五段・ワ行促音便/連用タ接続
て	テ	助詞-接続助詞
から	カラ	助詞-格助詞-一般
1	イチ	名詞-数
0	ゼロ	名詞-数
0	ゼロ	名詞-数
回	カイ	名詞-接尾-助数詞
以上	イジョウ	名詞-非自立-副詞可能
読ん	ヨン	動詞-自立/五段・マ行/連用タ接続
で	デ	助詞-接続助詞
いる	イル	動詞-非自立/一段/基本形
。	。	句点

EOS

図 1: 品詞体系変換例

[例 3]
```
動詞/子音動詞ワ行/タ系連用テ形<買って><カッテ>
 => 動詞-自立/五段・ワ行促音便/連用タ接続<買っ><カッ>,
    助詞-接続助詞<て><テ>
```

2.2 省略

変換規則は，形態素，読みを省略して定義することができる．例えば，品詞が” 普通名詞” である形態素をすべて，品詞” 名詞-一般” に変換したい場合には，例 4 のように記述する．

[例 4]
```
普通名詞 => 名詞-一般
```

形態素や読みが省略された変換規則を適用する場合には，変換後の形態素や読みは，変換前のものを利用する．

2.3 正規表現

変換規則の品詞，形態素，読みを正規表現により記述することができる．正規表現は，プログラミング言語 Perl Ver.5.x での記述方法に準拠している．例えば，(” 買って”，” カッテ”，” 動詞/子音動詞ワ行/タ系連用テ形”) の変換規則は，正規表現を使って例 5 のように記述できる．

[例 5]
```
動詞/子音動詞ワ行/タ系連用テ形<(.+) て><(+.) テ>
 => 動詞-自立/五段・ワ行促音便/連用タ接続<$1><$2>,
    助詞-接続助詞<て><テ>
```

この例のように，括弧”()” を用いて正規表現にマッチした文字列を記憶することができ，その記憶した文字列を変数”$数字” を使って右辺で利用することができる．先の変換規則を (” 買って”，” カッテ”，” 動詞/子音動詞ワ行/タ系連用テ形”) に適用した場合，変数 $1，$2 は，それぞれ，” 買っ”，” カッ” が代入され，二つの三組 (” 買っ”，” カッ”，” 動詞-自立/五段・ワ行促音便/連用タ接続”)，(” て”，” テ”，” 助詞-接続助詞”) に変換される．

2.4 再適用

通常，変換規則で変換した三組に対して，再度，変換規則を適用することはしない．しかし，再度，変換規則を適用しなければ，変換が実現できない場合などがある．そのため，ある変換規則で変換した三組に対して，再度，変換規則を適用する記述方法を用意した．右辺にある三組の中で，先頭に記号”@” が付いている三組は，再度変換規則を適用する．

$$P_1 < M_1 >< R_1 >, \ldots, P_m < M_m >< R_m >$$
$$\Rightarrow \quad \ldots, @P'_i < M'_i >< R'_i >, \ldots$$

例えば，(” １００”，” イチゼロゼロ”，” 数詞”) を三つの三組 (” １”，” イチ”，” 名詞-数”)，(” ０”，” ゼロ”，” 名詞-数”)，(” ０”，” ゼロ”，” 名詞-数”) に変換するように数字の連続を数字一文字づつに分割したい場合，再適用機能を用いて，次のような二つの規則を書けばよい．

```
[例 6]
   数詞<([0-9])([0-9]+)><(ゼロ|...|キュウ)(ゼロ|...|キュウ)+>
   => 名詞-数詞<$1><$3>,@数詞<$2><$4>

   数詞<[0-9]><(ゼロ|...|キュウ)>
   => 名詞-数詞
```

この例では，変換規則を再適用する三組は，品詞の変換を行っていない．このように右辺に記述する三組は，必ずしも変換後の品詞体系である必要はない．

2.5 変数

例 7 の場合，二つの規則は非常に似ているため一つの規則として記述したい．

```
[例 7]
   動詞/子音動詞ワ行/タ系連用テ形<(.+) て><(.+) テ>
   => 動詞-自立/五段・ワ行促音便/連用タ接続<$1><$2>,
      助詞-接続助詞<て><テ>

   動詞/子音動詞マ行/タ系連用テ形<(.+) で><(.+) デ>
   => 動詞-自立/五段・マ行/連用タ接続<$1><$2>,
      助詞-接続助詞<で><デ>
```

しかし，正規表現を用いて一般化するのは困難である．そこで，規則の一部を変数を用いること可能にした．変数”&数字” を用いて例 7’ のように記述することができる．

```
[例 7']
   {
     動詞/&1/タ系連用テ形<(.+)&2><(.+)&3>
     => 動詞-自立/\&4/連用タ接続<$1><$2>,
        助詞-接続助詞<\&2><\&3>
   }
   (子音動詞ワ行，て，テ，五段・ワ行促音便)
   (子音動詞マ行，で，デ，五段・マ行)
```

変数は，形態素，読み，品詞のどの部分でも記述可能である．また，括弧”” 内には，複数の規則を記述することができる．そのため，動詞の活用などをまとめて，記述することも可能である．(例 8)

```
[例 8]
   {
     &1/基本形 => &2/一段/基本形
     &1/未然形 => &2/一段/未然形
     ...
   }
   (母音動詞，動詞-自立)
   (動詞性接尾辞/母音動詞，動詞-接尾)
   ...
```

2.6 優先度

変換規則は，左辺に定義された三組が多いほど優先的に適用される．左辺に定義された三組の数が等しい適用可能な変換規則が複数あった場合には，先に定義された変換規則が優先される．

3 実験と考察

3.1 評価実験

Conbu を用いて，京都大学テキストコーパス，RWC コーパス (修正版)，東工大コーパス (EDR 版) の三つの品詞体系に対して 4 種類の変換規則を作成した．その 4 種類は，次の通りである．

- 京都大学テキストコーパス ⇒ RWC コーパス
- RWC コーパス ⇒ 京都大学テキストコーパス
- 京都大学テキストコーパス ⇒ 東工大コーパス (EDR)
- 東工大コーパス (EDR) ⇒ 京都大学テキストコーパス

形態素解析を用いて，この 4 種類の変換規則を評価した．まず，コーパスに付与された品詞体系と異なる品詞体系を扱う形態素解析器を利用して，コーパスを形態素解析を行う．次に，Conbu を用いて，コーパスに付与されている品詞体系へ変換する．そして，元のコーパスの形態素区切り，品詞，細品詞がどれだけ一致しているかにより評価した．評価実験に用いた形態素解析器は，茶筌[2] [6] と JUMAN[3] [5] を用いた．東工大コーパス (EDR 版) の品詞体系で形態素解析を行うことができる形態素解析器が見つからなかったため，EDR コーパスから京都大学テキストコーパスへの変換規則を評価できなかった．

評価尺度として，適合率，再現率を用いた．コーパスに含まれる形態素数を $Standard$，形態素解析器と Conbu により出力された形態素数を $System$，一致した形態素数を $Match$ とすると，適合率は，$Match/Standard$，再現率は，$Match/System$ で表す．さらに，形態素区切りが一致，名詞，助詞等の品詞が一致，動詞の活用形などを含む細品詞が一致の 3 種類の基準を設けた．

[2]茶筌は，Ver.2.3.3，辞書は，ipadic Ver.2.6.3 を使用．
[3]JUMAN は，Ver.4.0 を使用．

表 1: 評価実験結果

規則	規則数	コーパス	解析器	形態素		品詞		細品詞	
				適合率	再現率	適合率	再現率	適合率	再現率
京大 ⇒ RWC	512	RWC	JUMAN	92.2%	94.2%	90.8%	92.7%	83.1%	84.8%
RWC ⇒ 京大	458	京大	ChaSen	78.8%	87.3%	72.5%	80.3%	62.2%	68.9%
京大 ⇒ EDR	703	東工大	JUMAN	85.4%	90.6%	77.4%	82.1%	71.5%	75.8%
EDR ⇒ 京大	660	-	-	-	-	-	-	-	-
		京大	JUMAN	95.8%	97.7%	94.2%	95.9%	92.6%	94.2%
		RWC	ChaSen	95.6%	99.1%	95.3%	98.8%	81.0%	84.0%

評価実験結果を表 1 に示す．さらに，参考として，京都大学テキストコーパスを JUMAN で，RWC コーパスを茶筌で形態素解析した場合の適合率と再現率を記載した．

3.2 考察

実験結果より，細品詞の適合率，再現率は，約 60%から 85%の間であるため，自動的に品詞体系を変換するツールとしては，十分な変換精度ではない．さらなる変換規則の改善が必要であることがわかった．

評価対象となった 3 種類の変換規則の中で，京大 ⇒ *RWC* が特に適合率，再現率が良いのは，現在，この変換規則を中心に改善を行っているためである．

京都大学テキストコーパスと RWC コーパスに採用されている品詞体系は，非常に類似点が多いため，比較的変換規則を作成しやすい．一方，東工大コーパス (EDR) と京都大学テキストコーパスの品詞体系は，類似点が少なく，変換規則を作成しにくい．変換規則の記述方法を拡張する必要があるかもしれない．

また，人手によりすべての変換規則を記述するのは，非常に大きな労力が必要であった．そのため，変換ミスを発見し，自動的，または，半自動的に変換規則を作成するツールの開発が望まれる．

4 まとめ

本論文では，品詞タグ付きコーパスの品詞体系を変換するツール Conbu について述べた．Conbu は，規則ベースにより，品詞を変換する．京都大学テキストコーパス，RWC コーパス，東工大コーパス EDR 版を用いて，4 種類の変換規則を作成した．そして，3 種類の変換規則について評価実験を行い，適合率，再現率ともに約 60%から 85%で品詞体系を変換可能であることを示した．

評価実験の結果から，品詞体系を変換したコーパスにまだたくさんの変換ミスがあることがわかった．さらなく変換精度の向上が今後の課題である．また，人手により変換規則を作成するには，非常に多大な労力を必要とするため，変換ミスの発見，変換規則の自動，半自動的な作成手法の開発が必要である．

謝辞

本研究の一部は，文部科学省 21 世紀 COE プログラム「大規模知識資源の体系化と活用基盤構築」の支援を受けて行われました．

さらに，RWC コーパスの修正版を提供していただいた奈良先端科学技術大学院大学の松本裕治先生，ツールの開発支援していただいた株式会社ワーズビークルの渕武志氏の両氏に誠に感謝いたします．

参考文献

[1] Koichi Hashida, Hitoshi Isahara, Takenobu Tokunaga, Minako Hashimoto, Shiho Ogino, and Wakako Kashino. The RWC Text Database. In *LREC '98*, 1998.

[2] 乾健太郎, 脇川浩和. 品詞タグつきコーパスにおける品詞体系の変換. 情報処理学会研究報告 NL-132, pp. 87–94, 1999.

[3] 松田寛, 桐山和久, 山田悟史, 吉野圭一, 松本裕治. 部分形態素解析を用いたコーパスの品詞体系変換. 情報処理学会研究報告 NL-134, pp. 23–30, 1999.

[4] 東京大学 大学院情報理工学系研究科 電子情報学専攻 西田・黒橋研究室. 京都大学テキストコーパス Ver.3.0. http://www.kc.t.u-tokyo.ac.jp/nl-resource/corpus.html.

[5] 東京大学 大学院情報理工学系研究科 電子情報学専攻 西田・黒橋研究室. 日本語形態素解析システム JUMAN Ver.4.0. http://www.kc.t.u-tokyo.ac.jp/nl-resource/juman.html.

[6] 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. 形態素解析システム『茶筌』 Ver.2.3.3. http://chasen.aist-nara.ac.jp/hiki/ChaSen/.

[7] 野呂智哉, 橋本泰一, 徳永健伸, 田中穂積. 大規模日本語文法の開発. 自然言語処理, Vol. 12, No. 1, pp. 3–32, 2005.