# UniDic2: 拡張性と応用可能性にとんだ電子化辞書

小木曽 智信
togiso@ninjal.ac.jp
国立国語研究所・奈良先端科学技術大学院大学

伝 康晴
den@ cogsci.l.chiba-u.ac.jp
千葉大学

## 1. はじめに

発表者らは、形態素解析辞書UniDicの開発・公開を行ってきた。UniDicは『現代日本語書き言葉均衡コーパス』(BCCWJ)への形態論情報付与に用いられたほか、形態素解析辞書としてさまざまな形で利用されている。2012年末には、MeCab用辞書UniDic-mecabをオープンソースの完全なフリーソフトウェアとして公開している[1]。

一方、UniDicは当初より、形態素解析辞書以外の利用を視野に入れた設計がなされている(伝ほか2007)。多様な情報が付加されたUniDicの見出し語は、自然言語処理、音声処理、言語研究等の各方面で利用が見込まれる価値の高いデータである。UniDicのこうした可能性を引き出すためには、言語資源として利用しやすい形式でデータを提供する必要がある。

そこで、UniDicの見出し階層構造を反映したXML形式で辞書データを作成した。あわせて、このデータをもとにしてユーザがカスタマイズした辞書データベースを作り、形態素解析辞書を作成することのできるユーティリティツール群を作成した。これらをUniDic2として公開する。本発表は、このUniDic2の設計と実装について述べる。

## 2. UniDicの概要

UniDicは斉一な単位による解析を実現するために、見出し語の認定を厳密なルールによって定めた「短単位」を見出し語に採用している(小椋ほか2011)。さらに、柔軟な見出し語付与を可能にするために、見出し語に語彙素・語形・書字形・発音形という階層構造を持たせ、表記の揺れや語形の変異にかかわらず同一の見出しを与えることを可能にしている(図1)。個々の見出し語には、語種やアクセント型などの豊富な情報が付与されている。

UniDicの見出し語を管理する形態論情報データベース上では、語彙素・語形・書字形・発音形の階層構造をそのままテーブル構造に反映させ、各表を関連づけて見出し語を格納している(小木曽ほか2011)。語彙素に約23万項目、語形に約26万項目、書字形に約42万項目の見出し語を収録している[2]。


**図1　UniDic見出し語の階層構造**

これら4つの表は、語形変化表(語頭・語末変化表および活用表)と組み合わせて形態素解析辞書の見出し語表にまで展開される。すなわち、表中の個々の見出し語は、語頭変化・語末変化・活用変化を経て出現形(表層形)が派生される。

従来のUniDic(1.2系列)では、こうしてできる展開後の語彙表に、コーパスから学習したコストを付したものを配布していた。語彙表には、UniDicが持つ豊富な情報をすべて出力するため、一つの出現形が持つ属性は25に及んだ。

## 3. UniDic2

### 3.1. UniDic2のコンセプト

UniDic2では、見出し語データの可読性を高めるとともに、ユーザによるカスタマイズを可能にするために、UniDic見出し語の階層構造をそのまま反映させたXML形式で見出し語データ(基本形語彙表)を提供する。語彙表に出力する情報は必要な最小限の属性(基本情報)とし、それ以外の情報(付加情報)は別ファイルに区別した。また、基本形語彙表とともに活用表などの語形変化表も公開し、出現形までの展開をユーザの手元で行えるようにした。

このように、辞書見出し語の提供方式を整理したことにより、形態素解析以外の目的でも見出し語データを利用しやすくなっている。

形態素解析辞書としてのUniDicは、上述の基本形語彙表を出現形レベルまで展開したものに、形態素解析器MeCabでコーパスから学習した単語コス

[1] http://sourceforge.jp/projects/unidic/

[2] 2013年1月現在。古文用の見出し語を含む。

トを付与したものである。コストは、従来の UniDic と同様、BCCWJ のコアデータを中心とするコーパスから学習している。この基本辞書に出力される情報は、各階層で見出し語を区別し、語形変化を行うのに必要な最小限の情報と、機械学習に利用している素性（「語種」）のみから成る。アクセント型や仮名形はこの中に入らないため、基本辞書には出力されない。後述のツールを用いることで、この基本辞書に必要に応じて必要な属性を追加して、これを MeCab 用のソース辞書とすることができる。

なお、公開中のオープンソース版 UniDic-mecab は上記の基本辞書相当のものであるが、利用者の多いアクセント型・仮名形を追加したものも追加公開している。

### 3.2. UniDic XML

UniDic2 で提供される、見出し語の階層構造をそのまま表現した XML 形式のデータである基本形語彙表（lexBaseCore.xml）は、基本辞書に相当する次の属性から構成されている。

■語彙素レベル
語彙素読み(lForm)・語彙素表記(lemma)・類(class)・語種(goshu)(・語彙素細分類(subLemma))
■語形レベル
語形基本形(formBase)・語形代表表記基本形(formOrthBase)・品詞(pos)(・活用型(cType)・活用型細分類(subCType))(・語頭変化型(iType)・語末変化型(fType))
■書字形レベル
書字形基本形(orthBase)・仮名形基本形(kanaBase)
■発音形レベル
発音形基本形(pronBase)

例として、語彙素「ヤハリ（矢張り）」の一部を示す。

```
<Lemma lemma="矢張り" lForm="ヤハリ" class="相" goshu="和">
  <Form formBase="ヤハリ" formOrthBase="やはり" pos="副詞">
    <Orth orthBase="やはり" kanaBase="ヤハリ" />
    <Orth orthBase="ヤハリ" kanaBase="ヤハリ" />
    <Orth orthBase="矢張り" kanaBase="ヤハリ" />
    <Pron pronBase="ヤハリ" />
  </Form>
  <Form formBase="ヤッパリ" formOrthBase="やっぱり" pos="副詞">
    <Orth orthBase="やっぱり" kanaBase="ヤッパリ" />
    <Orth orthBase="ヤッパリ" kanaBase="ヤッパリ" />
    <Pron pronBase="ヤッパリ" />
  </Form>
(中略)
</Lemma>
```

このデータを元に、ユーザは、見出し語の追加や新規情報の付与を、各見出し語階層で自由に行うことができる。たとえば上記の例で、語形「ヤッパリ」の書字形として「矢っ張り」を追加する場合には、Form 要素（@formBase="ヤッパリ"）の子要素として Orth 要素(@orthBase="矢っ張り")を追加すればよい。

なお、新たな情報として、たとえば語彙素レベルで分類語彙表番号を付与したい場合、Lemma 要素に新しい属性を付ければよいわけだが、形態素解析辞書にこうした基本情報以外の属性を出力する場合には、後述する付加情報ファイルに記述する仕様となっている。

UniDic2 では、活用形展開をユーザの元で行えるため、活用語を新規に追加する場合には、辞書の見出し語（基本形＝終止形）を追加して適切な活用型を指定すれば、各活用形を自動で出力することができる。

なお、ある活用型のうち特定の語のみに存在する活用形（「歩っ（た）」）や活用語尾までカタカナ書きされた活用形（「アツイ」）など、通常の変化・活用表展開では出力できないものについては、「特殊変化・活用形（AltOrth）」として変化・活用後の形を基本形語彙表内に直接記述することができる。

活用以外の語形変化として語頭変化と語末変化がある。語頭変化は、連濁のように語頭が変化するもの（亀［語頭変化型=カ濁］：カメ，ガメ）、語末変化は促音化のように語末が変化するもの（三角［語末変化型=ク促］：サンカク，サンカッ）である。これらの変化は、語形につけられた語頭・語末変化型属性と、語頭・語末変化表とを組み合わせて生成される。

語頭変化表（iFormCore.xml）、語末変化表（fFormCore.xml）、活用表（inflCore.xml）も XML 形式で提供される。

## 4. UniDic Tools

以上のような XML ファイル群を活用するためのユーティリティとして UniDic Tools を提供する。UniDic Tools は次のツール群から成る。

1. XML ファイル群で記述された形態論辞書から辞書データベースを作成するツール
2. 辞書データベースから形態素解析システム用辞書を生成するツール
3. 辞書データベースを検索するツール
4. 辞書データベースから情報を取得し、形態素解析済みのテキストに情報を付加するツール

以下、4.1～4.4 で各ツールについて概説する。

### 4.1. 辞書データベースの作成

XML ファイル群を直接操作して活用形展開等の処理を行うことは速度面で現実的でないため、UniDic Tools では、辞書をデータベースに格納したうえで処理を行っている。

XML ファイル群で記述された形態論辞書から SQLite のデータベースファイルを作成できる。この処理は標準的な configure && make コマンドで実行できる。辞書データベースの作成にかかる時間は一般的なノート PC で数分である。

### 4.2. 形態素解析辞書の作成

辞書データベースから形態素解析システム MeCab 用の辞書を作成できる。後述するとおり、設定ファイルを切り替えることで、ユーザが指定したさまざまな情報を含む形態素解析辞書を作成することができる。

### 4.3. 辞書データベースの検索

辞書データベースを検索するための CUI および GUI ベースのツールを提供している。CUI 版検索ツールは Perl スクリプトで実装され、Linux および Windows のコマンドシェルで動作する。たとえば、以下のような検索が可能である。GUI 版は Windows 版だけが提供される。

```
# 語彙素が「痛い」のエントリーで活用形が「連用形-一般」のものを検索
> lemma="痛い" and cForm="連用形-一般"
イタイ|痛い|*|形容詞-一般|形容詞|連用形-一般|いたい|いたく|イタイ|イタク|和
イタイ|痛い|*|形容詞-一般|形容詞|連用形-一般|イタい|イタく|イタイ|イタク|和
イタイ|痛い|*|形容詞-一般|形容詞|連用形-一般|痛い|痛く|イタイ|イタク|和
```

### 4.4. 形態素解析済みテキストへの付加情報付与

基本辞書によって形態素解析を施したテキストに対して、後から様々な付加情報を付与するツールを提供している。

入力テキスト中の属性列の並びや出力テキストに含める属性列の並びなどは定義ファイル中に指定でき、柔軟な利用が可能である。

## 5. UniDicのカスタマイズ

### 5.1. 形態素解析辞書作成の流れ

UniDic2 の最大の特徴は、設定ファイルを切り替えることで、さまざまな情報を含む辞書データベースを作成できることである。たとえば、形態素解析システムを動作させる上で必要最小限の情報のみを含む辞書を作成することもできるし、アクセント型を付与する後処理システムで参照するためのアクセント情報を含む辞書を作成することもできる。さらに、ユーザが独自に語彙を追加したり、付加情報を追加したりすることもできる。

図 2 に XML ファイル群で記述された形態論辞書から辞書データベースを作成する過程の一例を図示する。この例では、基本情報（基本形語彙表・語頭変化表・語末変化表・活用表）を記述した XML ファイル群と、付加情報（語彙表発音形付加情報・活用表発音形付加情報）を記述した XML ファイル群とを辞書データベースに読み込み、基本形語彙表（lexBase）と変化形を展開処理した語彙表（lex）を作成する。

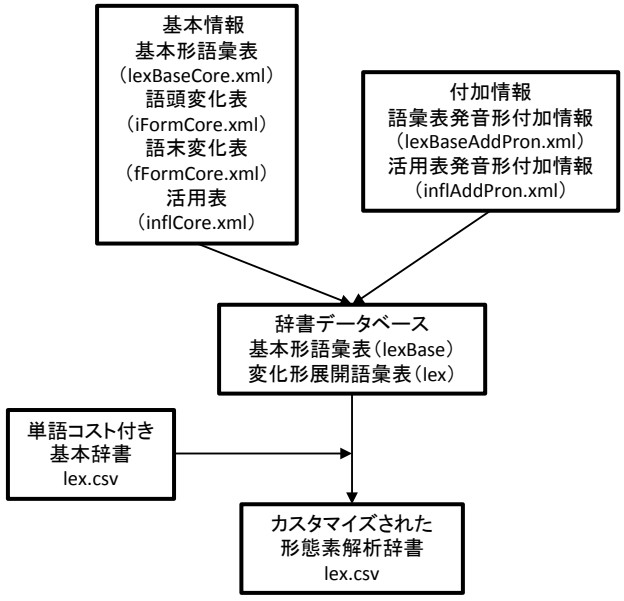


**図 2 辞書データベースの作成過程**

さらに、あらかじめ MeCab による学習で単語コスト情報が付与された基本辞書ファイル（lex.csv）からコスト情報を読み込み、lex に単語コストを付与することで、カスタマイズされた形態素解析辞書が出力される。なお、ユーザが新規追加した語のコストは、MeCab 0.994 で実装された単語コスト自動推定機能を使って付与する（CRF モデルファイルを利用する）。この一連の過程が 4.1 項・4.2 項のツールにより全自動で実行される。

この処理内容は、設定ファイル（*.def）に定義されており、これを書き換えることで、必要な情報を盛り込んだ形態素解析辞書を作成できる。以下、設定ファイルの仕様を説明する。

### 5.2. 付加情報の利用

形態素解析システム MeCab を動作させる上で必要最小限の情報のみを含む辞書データベースを作成するには、以下の内容の設定ファイルを用いる。

```
xmldir=xml_samples # 形態論辞書 XML ファイル群があるディレクトリ
lexBaseCore=lexBaseCore.xml # 基本形語彙表
iFormCore=iFormCore.xml   # 語頭変化表
fFormCore=fFormCore.xml   # 語末変化表
inflCore=inflCore.xml     # 活用表
```

さらに発音形付加情報を辞書データベースに含めるには、設定ファイルに以下の定義を追加する。

```
lexBaseAddPron=lexBaseAddPron.xml/aType:t,aConType:t   # 語彙表発音形付加情報を記述した XML ファイルとそこでの属性の名前と型
inflAddPron=inflAddPron.xml/aModType:t   # 活用表発音形付加情報を記述した XML ファイルとそこでの属性の名前と型
```

ここでは、lexBaseAddPron.xml 中でテキスト型の属性 aType（アクセント型）と aConType（アクセント結合型）が記述され、inflAddPron.xml 中でテキスト型の属性 aModType（アクセント修飾型）が記述されていることを宣言している。属性の型としては、テキスト型（t）・整数型（i）・実数型（r）が利用できる。

### 5.3. ユーザ定義辞書の利用

システム辞書の語彙をユーザが独自に定義した語彙で拡張することもできる。以下の設定ファイルでは、基本形語彙表を記述する XML ファイルとして、userCore.xml を追加で指定し、それらに対する発音形付加情報が userAddPron.xml に記述されていることを宣言している。

```
lexBaseCore=lexBaseCore.xml,userCore.xml
        # 基本形語彙表は 2 つの XML ファイルで記述
lexBaseAddPron=lexBaseAddPron.xml,userAddPron.xml/aType:t,aConType:t
        # それぞれの基本形語彙表に対する語彙表発音形付加情報ファイル
```

同様の仕組みにより、lexBaseCore で指定する XML ファイルの組み合わせをさまざまに変更することで、話し言葉用・近代語用・ブログ解析用など、目的に応じてカスタマイズされた語彙表を作成することが可能になる。

### 5.4. ユーザ定義付加情報の利用

語彙の拡張に加え付加情報も拡張できる。以下の設定ファイルは、発音形付加情報として新たに整数型属性 nMorae（モーラ数）を指定する場合である。

```
lexBaseAddPron1=lexBaseAddPron.xml,userAddPron.xml/aType:t,aConType:t
lexBaseAddPron2=lexBaseAddPron2.xml,userAddPron2.xml/nMorae:i
lexBaseAddPron="$lexBaseAddPron1;$lexBaseAddPron2"      # 2 種類の語彙表発音形付加情報を宣言し、; 区切りで連結
```

### 5.5. 形態素解析辞書に出力するフィールド

辞書データベースに取り込んだ属性を、形態素解析辞書に出力するためには、設定ファイルで次のように出力フィールドを記述する。

```
# 語頭・語末変化型・形と仮名形と語形を含む MeCab 辞書
addFields=iType,iForm,fType,fForm,kana,kanaBase,form,formBase
```

ここで指定したフィールドは、標準で出力される前述の基本辞書の属性の後に続いて出力される。

以上のように、語彙・付加情報をユーザが拡張できることで、極めてカスタマイズ性に富んだ辞書データベース（および形態素解析辞書）を作成することが可能になっている。

## 6. おわりに

UniDic2 により UniDic は今まで以上に利用しやすい言語資源として公開されることとなった。これにより UniDic のもつ可能性が引き出され、さらに広く多様な目的で利用されるようになることを期待したい。

## 参考文献


伝康晴・小木曽智信・小椋秀樹・山田篤・峯松信明・内元清貴・小磯花絵（2007）「コーパス日本語学のための言語資源：形態素解析用電子化辞書の開発とその応用」『日本語科学』22　pp.101-123

小椋秀樹・小磯花絵・冨士池優美・宮内左夜香・小西光・原裕（2011）国立国語研究所内部報告書『『現代日本語書き言葉均衡コーパス』形態論情報規程集第 4 版（上・下）』LR-CCG-10-05

小木曽智信・中村壮範（2011）　国立国語研究所内部報告書『『現代日本語書き言葉均衡コーパス』形態論情報データベースの設計と実装 改訂版』LR-CCG-10-06