日本語KWIC索引生成ソフトウェア KWIC

田野村 忠温

（2007年9月7日刊行、2015年6月11日最終更新、2018年5月3日微調整）

コーパス利用の普及のためには簡単に使えるコーパス処理ソフトウェアの存在が不可欠だと思われるが、残念ながら日本語に関しては研究者が手持ちの電子テキストをそのまま使って簡便にKWIC索引を生成することのできるソフトウェアがなかった。

そうしたことから、2007年に『コーパス日本語学ガイドブック』（末尾の付記参照）を刊行したときWindows上で作動する日本語KWIC索引生成ソフトウェアを試作してみた。初めて使うRubyによって短期間で書いた素朴なもので、遠からず高機能で使いやすいKWICソフトウェアが作られて普及するものと予想していたが、作成から10年以上経つ今も多くの方に使われている。サーチエンジンで“KWIC”を検索すると本ソフトウェアが日本語研究用のKWICソフトウェアとしては最上位のランクに表示される。これは作った甲斐があったという意味では喜ばしいが、コーパス日本語研究の水準の反映と受け止めれば寂しい現実である。

本ソフトウェアは上記ガイドブック刊行後も多少の機能強化を図った。そして、幸い現在のWindows上でも使えるようである。以下では、最新版の機能・用法と、検索結果の利用に関する簡単なヒントについて述べる。

内容

※しおりの ⊞ をクリックすれば目次を展開することができます。

※2018年5月の微調整版は2015年6月の版とほぼ同等で、差し替える意味はありません。

1　概要

本ソフトウェア KWIC は、日本語のテキストファイルから語句を検索し、見つかった用例をソート（並べ替え）された KWIC 形式で出力する。エクセルがインストールされていれば、検索結果をエクセルファイルにも出力する。

当初 jKWIC とか tKWIC といった名前にしようかとも思ったが、基本的にコマンドプロンプトで使うソフトウェアであることから、使用時の打鍵の手間を節約するために KWIC という名前にした。KWIC は次のような特徴を備えている。

○任意のテキストをそのまま検索することができる。
○テキストが桁折りされていても（＝改行をまたいだ語句も）検索できる。
○指定のディレクトリ（複数可）に含まれるすべてまたは一部のテキストを一括して検索する。
○複数の検索条件を記述したファイルに基づいて一気にまとめて検索できる。
○検索文字列に加えて前後の文脈の条件を指定可能で、いずれにも正規表現が使える。
○仮名１文字、漢字１文字、特定の五段活用動詞の全活用形といった条件を簡単に指定できる。
○用例は前後の文脈などに従ってソートした形で出力する。
○検索結果をエクセルファイルの形で出力し、自動的にエクセルで開くことができる。

2　インストールと使用の準備

2.1　KWICのインストール

(a) KWIC 関連ファイル一式のインストール（コピー）

「日本語 KWIC 索引生成ソフトウェア KWIC」のウェブページ（http://www.tanomura.com/research/KWIC/）に KWIC の最新版が置いてある。そこで説明している手順に従ってインストールする。

インストールと言ってもディレクトリ（フォルダ）を作成してそこに必要なファイルを数個コピーするだけである。アンインストールするには単に当該のディレクトリを中身ごと消せばよい。

※以下の説明では、KWIC を C:¥KWIC（＝ドライブ C の KWIC というディレクトリ）にインストールしたものとして話を進める。他のドライブ・ディレクトリにインストールした場合は説明を適宜読み替える必要がある。特に不都合のない限り C:¥KWIC へのインストールを推奨する。
※「¥」は英語、中国語などの Windows では「\」（back slash、逆斜線）、韓国語の Windows では「₩」になる。
※KWIC は日本語環境での使用を前提としている。日本語版以外の Windows で KWIC を使うときは上記のページで説明している設定変更を臨時に行う必要がある。

(b) デスクトップアイコンの作成

KWIC は基本的にコマンドプロンプトで実行するソフトウェアである。コマンドプロンプトでは各種の作業を合理的に行うことができるが、それには若干の知識を必要とする。そこで、KWIC を一般的な Windows アプリケーションソフトウェアのようにデスクトップから起動・実行するための仕組みを用意してある。

※KWIC をコマンドプロンプトで実行する方法については補説Ｃを参照。
※マニュアルの構成の変更に伴い、従来の RunKWIC を KWIC に改称した。

KWIC を簡単に使えるよう、次の手順によってデスクトップ上に KWIC へのショートカットアイコンを作成する。手順は環境によって多少異なる可能性がある。

・Windows XP/Vista/7 の場合
(1) マイコンピュータ（Windows Vista/7 ではコンピュータ）（またはエクスプローラ）で C:¥KWIC を開く。
(2) その中にある KWIC という名前の３つのファイルのうち「HTML Application（または、HTML アプリケーション)」という説明の付いているもののアイコンを右クリックし、「送る(N)」→「デスクトップ（ショートカットを作成)」を実行する。
(3) これによりデスクトップに「KWIC へのショートカット」という名前のファイル（アイコン）が作られる。

・Windows 8/8.1 の場合
(1) コンピュータ（またはエクスプローラ）で C:¥KWIC を開く。
(2) その中にある KWIC という名前の３つのファイルのうち「HTML アプリケーション」という説明の付いているもののアイコンを右クリックし、「ショートカットを作成」を実行する。
(3) C:¥KWIC に「KWIC - ショートカット」という名前のファイルが作られるので、それをデスクトップに移動させる（アイコンとして表示される)。

※KWIC という名前の３つのファイルの完全な名前は KWIC.def 、KWIC.exe、KWIC.hta である。それぞれに「DEF ファイル」「アプリケーション」「HTML Application（HTML アプリケーション)」という説明が付いている。

アイコンの「KWIC へのショートカット」「KWIC - ショートカット」という名前は長くて不格好なので「KWIC」に変えればすっきりする。それには、アイコンを右クリックして「名前の変更(M)」を選択し、「 へのショートカット」「 - ショートカット」を削除してから[Enter]キーを押せばよい。

以上で KWIC のインストールは完了である。

2.2 コーパスの準備

コーパスはどこのディレクトリに置いてもかまわないが、将来コーパスのデータを増やす可能性を見越して、コーパスを収めるディレクトリを１つ用意し、その中にサブディレクトリ（階層的でもよい）を作って日本語テキストを分けて収めるようにするとよい。

※以下の説明では、コーパスを C:¥corpora に置くものとして話を進める。他のドライブ・ディレクトリを用いる場合は説明を適宜読み替える必要がある。特に不都合のない限り C:¥corpora の使用を推奨する。

※コーパスを置くディレクトリの名前はいわゆる半角文字である必要はなく、C:¥コーパス のような名前でも差し支えない。

日本語テキストの文字コードは Shift_JIS である必要がある。

KWIC をすぐに試せるよう、「日本語 KWIC 索引生成ソフトウェア KWIC」のページには青空文庫所収の文学作品約 770 件をまとめたデータを掲載してある。それをインストールすれば、C:¥corpora¥aozora というディレクトリが作られ、その中の作家別のサブディレクトリに文学作品のファイルがコピーされる。C:¥corpora というディレクトリが事前に存在しなければそれも同時に作られる。

※日本語テキストを C:¥corpora 以外のところに置きたい、あるいは、すでに置いてあるという場合は、C:¥KWIC にコピーされている定義ファイル KWIC.def をメモ帳かテキストエディタで読み込み、「パス＝～」という箇所を適宜書き換えて保存し直す。
※複数のコーパスを使い分ける方法は 7.2 で説明する。

3　用法(1) 簡易検索

3.1　簡易検索と連続検索

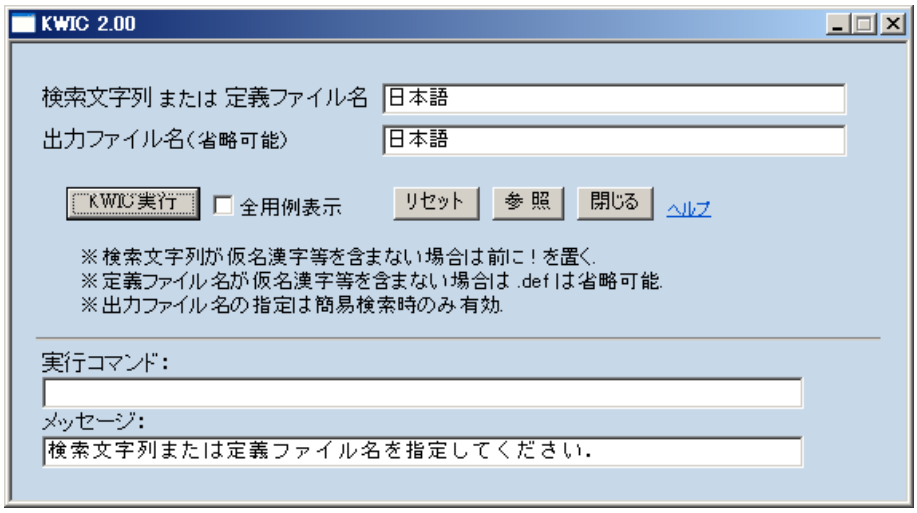
KWIC には簡易検索と連続検索の２通りの使い方がある。簡易検索では検索文字列を直接に指定して検索を行う。連続検索ではあらかじめファイルに複数の検索条件を記述しておき、それに基づいて一気に検索を行う。

3 節で簡易検索、次の 4 節で連続検索について説明する。

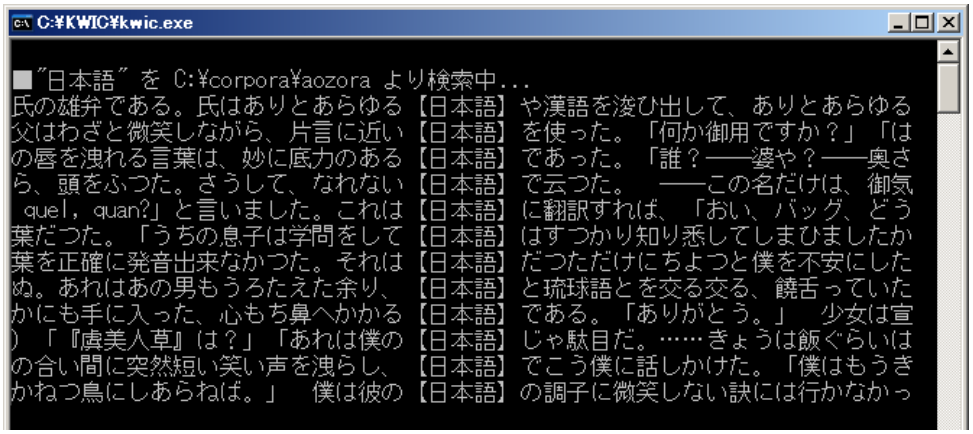
3.2　簡易検索の方法

デスクトップにある KWIC のアイコンをダブルクリックすれば、次の図のようなウィンドウが開く。例えば、コーパスから「日本語」という文字列を検索し、その結果を「日本語」という名前のエクセルファイルに出力するには、検索文字列と出力ファイル名の両方に「日本語」を指定して「KWIC 実行」のボタンを押す。

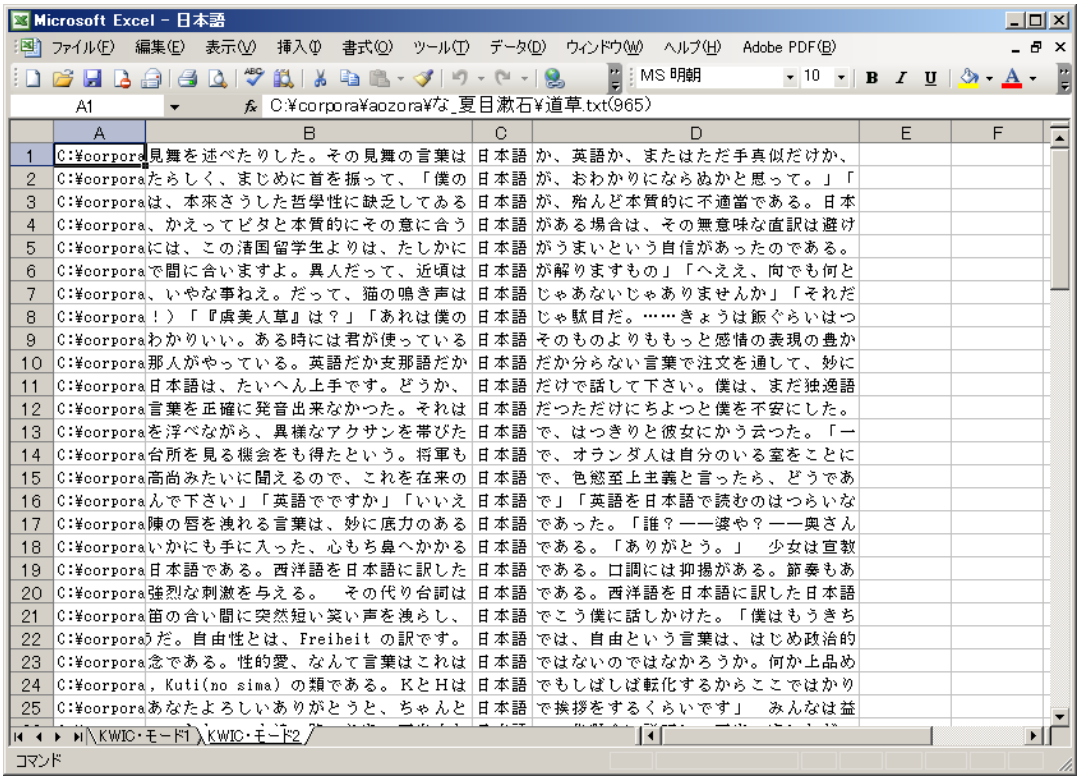
※出力ファイル名は後で見て検索内容が分かるものにする。検索文字列と一致させる必要は特にない。



検索が始まると、次のような黒い背景のコマンドプロンプトのウインドウが開き、用例が見つかるごとに表示される。検索文字列は「【 】」で囲んで表示される。



検索が終わると、検索結果は検索文字列の前後の文脈などに関してソートされた形でエクセルファイル 日本語.xls（または日本語.xlsx）に出力され、自動的にエクセルで開かれる。



検索終了後には、用例の件数などを示すポップアップメッセージが表示される。これは放っておいても数秒後に消える。また、コマンドプロンプトのウィンドウも自動的に閉じる。

検索結果のエクセルファイルには２つないし３つのシート「KWIC・モード 1」「KWIC・モード 2」（「KWIC・モード 3」）が作られている。各シートでは用例がそれぞれ異なる様式でソートされている。具体的には次の通りである。

モード	シートの内容
1	先行文脈に基づいてソートした検索結果
2	検索文字列＋後続文脈に基づいてソートした検索結果
3	後続文脈に基づいてソートした検索結果

エクセルファイルの検索文字列の列に１通りの文字列――上の例で言えば「日本語」――しか現れない場合にはモード 2 とモード 3 によるソートは同一の結果になるのでモード 3 のシートは作られない。

モード 3 のシートが作られる可能性があるのは、検索文字列の指定に正規表現を用いた場合である。KWIC では、検索文字列の指定には正規表現および KWIC 独自の擬似正規表現を使うことができる（正規表現については「日本語 KWIC 索引生成ソフトウェア KWIC」のページに掲載している正規表現の解説、擬似正規表現については本マニュアルの 6.1 を参照）。例えば、検索文字列に「探[さしすせそ]」という正規表現――「さしすせそ」を囲む角括弧は半角文字で入力する――、出力ファイル名に「探す」を指定して検索すれば、「探す」の全活用形の用例が得られる。この場合、検索結果のエクセルファイルの検索文字列の列には「探さ」「探す」「探し」「探せ」「探

そ」の最大５通りの文字列が現れる。実際に検索してみて３つのシートの内容を確認されたい。

※検索文字列の指定に正規表現を用いても、得られた検索結果において当該の部分が１通りの文字列だけである場合はモード３のシートは作られない。
※用例のソートは Shift_JIS 文字コードに基づいて行われる（文字コードについては「日本語 KWIC 索引生成ソフトウェア KWIC」のページにある解説文書を参照）。
※指定した出力ファイル名を name とすると、検索結果はエクセルファイル name.xls（または name.xlsx）だけでなく、name_1.tsv、name_2.tsv（、name_3.tsv）という２つないし３つのタブ区切り形式のテキストファイルにも出力される。それぞれ、エクセルファイルの２つないし３つのシートに対応している。これらのテキストファイルは、検索結果をほかのソフトウェアで読み込んで処理する場合や、検索結果の件数が多くてエクセルに格納できなかった場合などに利用する。また、検索文字列や見つかった用例の件数などの情報が name_0.tsv というテキストファイルに記録される。これは実際にはタブ区切り形式でないが、拡張子（ファイル名の最後のピリオド以後の部分）を name_1.tsv その他に揃えている。
※検索文字列が見つからなかった場合は、接尾辞「_0」のテキストファイルだけが作成され、エクセルファイルおよび接尾辞「_1」「_2」「_3」のテキストファイルは作成されない。処理が正常に行われた場合、１度の検索において作成されるテキストファイルの数は、４つ（「_0」「_1」「_2」「_3」）、３つ（「_0」「_1」「_2」）、１つ（「_0」だけ）の３通りの可能性があることになる。
※KWIC の主なチェックボックス、ボタン、リンクの機能は次の通りである。
「全用例表示」チェックボックス：
通常は 100 件を超えた検索結果の画面への出力は 10 件ごとに行われるが、「全用例表示」チェックボックスにチェックを入れておくとすべての結果が画面に出力される。画面出力には多少時間がかかるので、用例が多いときは画面出力を間引くほうが処理速度の点で有利である。全用例表示のチェックの有無にかかわらず、ファイルの形で得られる検索結果は同一である。
「参照」ボタン：
KWIC のディレクトリを開く。定義ファイルや検索結果のファイルを参照するときなどに使う。
「ヘルプ」リンク：
Internet Explorer で「日本語 KWIC 索引生成ソフトウェア KWIC」のページを開く。

3.3 出力ファイル名の指定省略時の出力ファイル名

出力ファイル名の指定は省くことができる。

その場合、出力ファイル名は自動的に設定される。定義ファイル KWIC.def――この後すぐ 3.4 で説明する――の中で出力ファイル名が記述されていればそれに３桁の連番を加えたもの、定義ファイルに出力ファイル名が記述されていなければ kwic という名前に３桁の連番を加えたものになる。例えば、name に連番が加わると name001 のようになる。

テキストファイルについては連番の後ろにさらに接尾辞「_0」「_1」「_2」「_3」が付加されるので、実際のファイル名は name001_0.tsv、name001_1.tsv、name001_2.tsv、name001_3.tsv のようになる。

簡易検索時の出力ファイル名の決まり方を整理しておくと、名前は

(1) 明示的に指定されたファイル名
(2) 定義ファイル KWIC.def で指定されたファイル名＋３桁の連番
(3) kwic＋３桁の連番

という優先順位で決まり、テキストファイルではその後ろに接尾辞「_0」「_1」「_2」「_3」が加えられることになる（連番の決定方法の詳細については 7.2 の「簡易検索時の出力ファイル名に加えられる連番の決定方法」を参照）。

※１つや２つの検索を試してみるときは出力ファイル名の指定を省いても問題ないが、多数の検索を試すときや実際の研究に関わる検索を行うときは出力ファイル名を明示的に指定すべきである。そうしなければ、後になってそれぞれのファイルをいちいち開いてみないことにはそれがどのような検索の結果であるか分からなくなってしまう。

3.4 定義ファイル

上の説明に出てきた定義ファイル KWIC.def は次のような内容のファイルである。各行で検索に関わる諸条件を指定している。日本語テキストの置かれた場所を指定する「パス＝～」の記述は必須であるが、残りの行の記述はなくてもかまわない。定義ファイルの書き方および各条件の詳細は 5 節で述べる。

```
# KWIC 定義ファイル・簡易検索        ← 「#」で始まる行は単なる注釈

パス＝C:¥corpora                    ← 日本語テキストの所在（必須）
出力＝検索結果                      ← 出力ファイル名（任意）
タグ＝前置                          ← タグの位置（任意）
行長＝80                            ← 行長（任意）
```

上の KWIC.def では C:¥corpora のディレクトリに含まれるすべてのテキストを検索する。ほかの場所にあるテキストを検索する場合や、C:¥corpora の中にあるサブディレクトリにあるテキストだけを検索する場合などは、「パス＝～」の定義を適宜書き換える。

4 用法(2) 連続検索

4.1 連続検索の方法――最も単純な例

連続検索では、検索文字列や出力ファイル名を画面上で直接に指定するのではなく、それらの情報を書いた連続検索用の定義ファイルを事前に用意しておき、その定義ファイルの名前を指定して検索を行う。この方法を使えば、複数通りの検索を自動的に連続して行うことができる。

インストールと同時に C:¥KWIC ディレクトリには連続検索の定義ファイルのサンプルが４つ（sample1.def～sample4.def）コピーされている。

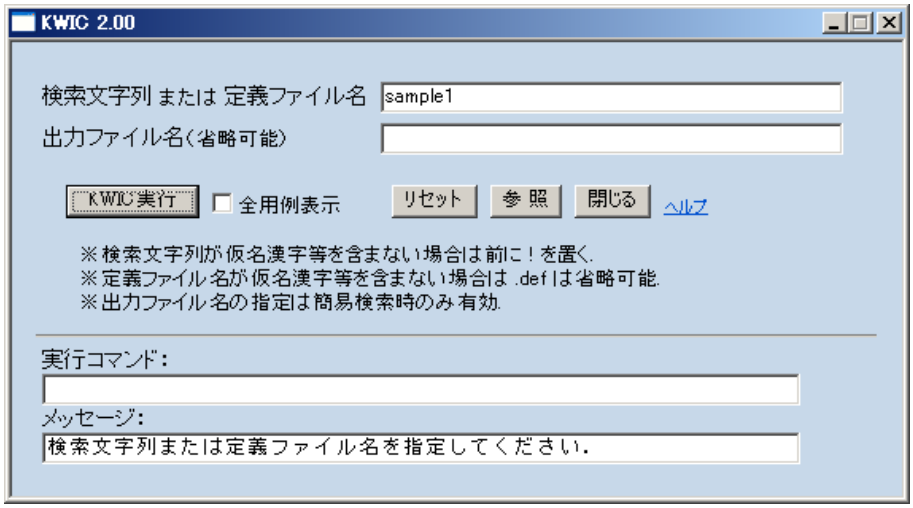
まず、sample1.def は次のような内容である。この例は検索を１度だけ行う。

```
# KWIC 定義ファイル・連続検索１

パス＝C:¥corpora¥aozora
出力＝日本語                        ← 出力ファイル名を指定
検索＝日本語                        ← 検索文字列を指定して検索を実行
```

この定義ファイルを使って連続検索を行うには次の図のように定義ファイル名に sample1.def

を指定して「KWIC 実行」ボタンを押す。実際には.def は省いて sample1 と指定するだけでよい。



sample1.def の内容は、ディレクトリ C:¥corpora¥aozora の中にあるすべてのテキストから「日本語」という文字列を検索し、その結果を「日本語」という名前のファイルに出力することを意味する。これの実行結果は、(C:¥corpora に含まれるコーパスが aozora だけであれば) 簡易検索の最初の例 (3.2) の場合と同じである。簡易検索の場合と同様にエクセルファイルとテキストファイルが作られる。

4.2 複数の検索の連続実行の例

連続検索では、複数の検索をまとめて行うことができる。定義ファイルのサンプル sample2.def は次のような内容である。

```
# KWIC 定義ファイル・連続検索 2

パス＝C:¥corpora¥aozora

出力＝よほど
検索＝よほど|よっぽど|余程          ← １度目の検索

出力＝さがす
検索＝(探|捜|さが)[さしすせそ]      ← ２度目の検索
```

実行方法は sample1.def の場合と共通である。この定義ファイルを使うと、２種類の検索が順次実行され、その結果がそれぞれ指定の名前のエクセルファイルに出力される。この例では検索文字列の指定に正規表現を使用しており、１度目の検索では「よほど」「よっぽど」「余程」の用例、２度目の検索では「探す」「捜す」「さがす」の全活用形の用例が検索される。

4.3 さらに進んだ使い方

C:¥KWIC にはほかにも sample3.def、sample4.def という定義ファイルのサンプルが収められている。これまでの例には出て来なかった使い方や使いこなしのヒントを紹介するものである。それぞれを実行し、結果を定義ファイルの内容と説明および次節の解説と照らし合わせて確認されたい。

5 定義ファイルの詳細

5.1 定義ファイルの名前

簡易検索時に使う定義ファイルは KWIC.def という名前にし、KWIC.exe と同じディレクトリに置く。

連続検索に使う定義ファイルの名前は基本的には自由で、拡張子についても、.def を付けてもよいし、その他の任意の拡張子を付けてもよいし、拡張子を付けなくてもよい。ただし、条件が１つあり、ファイル名が以下の文字――日本語の検索において特に重要な文字――を含むときは、必ず拡張子.def を加えなければならない。これは、定義ファイル名のつもりで指定したものが検索文字列と誤認されるのを防ぐための措置である。

(1) 全角文字の仮名・漢字
(2) 正規表現に使う半角文字の記号のうち + ( ) [ ] { }

定義ファイル名がこれらの文字を含まず、かつ、拡張子が.def である場合は、定義ファイル名指定時に拡張子を省いて name.def の代わりに単に name と指定すればよい。

※name と name.def の両方が存在する状況で name と指定したときは、拡張子なしの name が優先的に使用される。

連続検索時の定義ファイルは任意の場所に置くことができる（定義ファイルをカレントドライブ・カレントディレクトリ以外の場所に置くときのファイル名の指定方法については 7.2 の「出力ファイル名および定義ファイル名のパス指定」を参照)。

5.2 記述できる行の種類

定義ファイルには次の３種類の行を記述することができる。

(1) 注釈行・空行
(2) 設定行……パス(path)行、出力行、タグ(tag)行、行長行、通知行
(3) 命令行……検索行、終了行

注釈行（＝「#」「;」「'」のいずれかで始まる行）と空行は定義ファイルに自由に含めることができ、KWIC の処理に影響しない。

設定行は検索文字列以外の検索条件や結果の出力様式などを設定するものである。命令行は KWIC に動作の指示を与えるものである。簡易検索用の定義ファイルでは命令行は意味を持たず、記述してあっても無視される。

	行の種別	意味	必須・任意の別	省略時解釈
パス＝～	設定行	日本語テキストの所在	必須	—
出力＝～	設定行	出力ファイル名	任意／必須	kwic／—※1
タグ＝～	設定行	タグ付加の有無・位置	任意	前置
行長＝～	設定行	タグを含まない１行の長さ	任意	80
通知＝～	設定行	終了時の通知音声	任意※2	—
検索＝～	命令行	検索文字列を指定して検索	必須※3	—
終了	命令行	処理を終了	任意※3	—

※1　簡易検索時は任意、省略時解釈は kwic。連続検索時は必須。
※2　通知音声の再生に関しては Windows のバージョンによる事情の違いがある。5.4(e)の注を参照。
※3　連続検索時のみ。

5.3　記述の順序

簡易検索時の定義ファイルでは、４種類の設定行の記述の順序は自由である。設定行の記述に重複があるときは最後のものが有効になる。

連続検索の場合には、KWIC は定義ファイルを最初から１行ずつ読んでいき、検索行が現れるごとに検索を実行する。当該の検索に関わる設定行はその検索行よりも前に記述されている必要がある（設定行相互の順序は自由）。設定行の記述に重複があるときは検索行から見て最新（最後）のものが有効になる。

連続検索で複数の検索を行う場合、出力ファイルの重複を避けるために、それぞれの検索行に先立って相異なる出力ファイル名を設定する必要がある。出力ファイル名以外の情報は指定がなければ直前の検索で使われたものが引き継がれるので、変更の必要がなければ繰り返し指定する必要はない。

5.4　各行の記述の詳細

各種の行の記述の詳細は以下の通りである。

(a) パス行（必須）

検索対象とする日本語テキストの置かれたディレクトリの名前（例：「パス＝C:¥corpora」「パス＝C:¥corpora¥aozora」）または日本語テキストのファイルの名前（例：「パス＝C:¥corpora¥asahi¥asahi1987.txt」「パス＝C:¥corpora¥asahi¥asahi198?.txt」）を指定する。ディレクトリ名を指定した場合は、そのディレクトリ（およびその中にあるすべてのサブディレクトリ）にあるすべての日本語テキストが検索の対象とされる。

パス行には複数のディレクトリ名・ファイル名を半角の「|」で区切って指定することができる。例えば、「パス＝C:¥corpora¥asahi | C:¥corpora¥mainiti」のように指定すれば、asahi と mainiti の２つのコーパスがともに検索される。「|」の前後には半角の空白があってもよい。

※ディレクトリ名・ファイル名には、環境変数名を含めることができる。その場合、環境変数名は「%名前%」の形で指定する。例えば、「パス＝%corpora%:¥corpora¥aozora」のようにする。そして、パソコンごとに環境変数 corpora に C や D などの値を適宜設定する。これにより、ドライブ構成の異なる複数のパソコン間での定義ファイルの共有が可能になる。（環境変数の設定は、「コントロールパネル」→「システム」（または、「マイコンピュータ」→「プロパティ」）によって開く「システムのプロパティ」ダイアログの「詳細設定」タグにある「環境変数」ボタンを押して行う。「ユーザー環境変数」で「新規」を選び、「新しいユーザー変数」ダイアログで変数名に「corpora」、変数値に「C」などを指定する。）

(b) 出力行（連続検索時必須）

検索結果などを出力するファイルの名前を指定する。ファイル名には全角文字も使える。

指定されたファイル名が拡張子を持たない場合、および、xls（または xlsx）の拡張子を持つ場合は、検索結果はエクセルファイルおよびタブ区切り形式のテキストファイルに出力される。ファイル名が xls（xlsx）以外の拡張子を持つ場合は、エクセルファイルへの出力は行わない。

出力テキストファイル名は、指定された名前（の拡張子を除いた部分）の後ろに次の１つないし２つの要素がその順に付加されたものになる。

(1) ３桁の連番（簡易検索時にコマンドラインで出力ファイル名を指定しなかった場合のみ）
(2) 接尾辞「_0」「_1」「_2」「_3」（すべての場合）

連番については3.4および7.2(「簡易検索時の出力ファイル名に加えられる連番の決定方法」)、接尾辞については 3.3 を併せて参照されたい。

指定された出力ファイル名がドライブ名やディレクトリ名を含まない場合は、出力ファイルはカレントドライブのカレントディレクトリに作成される。出力ファイルをほかの場所に作成したいときは、適宜ドライブ名やディレクトリ名を加えればよい（具体例については 7.2 の「出力ファイル名および定義ファイル名のパス指定」を参照）。

(c) タグ行

出力の各行にタグ（＝検索文字列がどのファイルの何行目で見つかったかを示す情報）を付加するかどうか、付加する場合はどこに加えるかを指定する。

「タグ＝前置」と指定すると、タグは当該行の冒頭に置かれる。「タグ＝後置」と指定すると、タグは当該行の末尾に置かれる。「タグ＝なし」と指定すると、タグは付加されない。

タグ行の指定を省略すると「タグ＝前置」が仮定される。

※タグは「ファイル名(行数)」の形をしており、例えば「C:¥corpora¥aozora¥な_夏目漱石¥吾輩は猫である.txt (123)」というタグは、当該の用例が C:¥corpora¥aozora¥な_夏目漱石¥吾輩は猫である.txt というファイル名のテキストの 123 行目にあることを示す。タグを用いることにより原文の当該箇所を容易に参照することができる（その方法については補説 A を参照）。

(d) 行長行

タグの部分を除く１行の長さを全角文字数を単位として指定する。

指定可能な行数の下限は 10、上限は特にない。

行長行の指定を省略すると「行長＝80」が仮定される。

※出力される KWIC 索引の行は指定された行長よりも 2 字短いものになる。これは、本ソフトウェアの初期のバージョンにはエクセルを扱う機能がなく、KWIC 索引を次のようなテキストとして出力していたことに由来している。以下は「行長＝40」を指定して出力した場合の例で、各行は検索文字列を囲む「【 】」の 2 字を含めて 40 字の長さになっている。

「いや、いや……」学生は手を振った。【余程】のショックを受けたらしく、唇を震わせ
何度も何度も繰り返して、口説いたのが【よほど】効いたのでしょう、義理のある養家を
ころへやって来ました。彼は、この箱が【よほど】気に入ったのか、さも面白く珍しそう
くらいなら、最初から取り合わない方が【よっぽど】ましだった。それで彼女にはどうし
して聞く者よりか喋舌ている連中の方が【余程】面白そうであった。　先ずこのがやがや
ってホテルに向って歩いてゆく彼の方が【よほど】気が気でなかった。そのうち彼はこり
の壜を持つて振つて見せた。中にはまだ【余程】這入つてゐた。梅子は手を敲いて洋盞を

(e) 通知行（Windows 出荷時の状態では Windows XP のみ）

処理終了時に再生する WAVE 形式の音声ファイルの名前を次のいずれかの形式で指定する。

(1) 通知＝name1.wav, name2.wav
(2) 通知＝name1.wav

(1)の形式で指定した場合、すべての検索が正常に終了したときは name1.wav、エラーで処理を中止したときは name2.wav が再生される（定義ファイルが見つからない、通知行の設定に誤りがあるなどの場合を除く）。(2)の形式で指定した場合、検索が正常に終了したときは name1.wav、エラーで処理を中止したときには C:¥WINDOWS¥Media¥chord.wav が存在すれば当該ファイル、存在しなければ name1.wav が再生される。

指定された音声ファイルの名前にドライブ名やディレクトリ名の指定がない場合は、その名前の音声ファイルをまずカレントドライブのカレントディレクトリで探し、そこで見つからなければ次に C:¥WINDOWS¥Media で探す。C:¥WINDOWS¥Media には Windows 標準の状態で chimes.wav、chord.wav、ding.wav、notify.wav、tada.wav などの音声ファイルが入っている。

通知行は、連続検索時の終了行よりも前であれば、定義ファイルのどこに記述してもよい。ただし、通知行を置くのであれば定義ファイルの冒頭に置くのがよい。そうすれば、定義ファイルの通知行以前の記述ミス（ただし、通知行の誤りを除く）による処理中止時に警告音を鳴らすことができる。

「通知＝name1.wav, name2.wav!」「通知＝name1.wav!」のように行末に「!」を付えて指定すると、音声ファイルが見つからなかった場合もエラーとせず処理を継続する。

通知行の指定を省略すると通知音声の再生は行わない。

※Windows Vista/7 ではそのままでは通知音声再生の機能が使用できず、音声を再生しようとするとエラーになる。これは、Windows XP の C:¥WINDOWS¥System32 にある sndrec32.exe というソフトウェアが Vista、7 では省かれたことによる。XP を持っている場合は sndrec32.exe を Vista、7 の同じ場所にコピーすれば通知音声の再生が可能になる。（管理者権限のないアカウントで使用する場合は、sndrec32.exe 初回使用時に 1 つの手順が必要かも知れない。インターネットで「windows 7 sndrec32」などのキーワードで検索すれば解説が得られる。）Windows 8/8.1 でも同様ではないかと思われるが、確認できていない。

(f) 検索行（連続検索時必須）

検索文字列を指定し、検索を行う。文字列の指定には正規表現が使える。

検索行には、検索文字列だけでなく、その前後の文脈の条件も記述することができる。文脈の条件は次の３通りの形式のいずれかで指定する。文脈の条件の指定にも正規表現が使える。

(1) 検索＝先行文脈【検索文字列】　　（先行文脈を指定）
(2) 検索＝【検索文字列】後続文脈　　（後続文脈を指定）
(3) 検索＝先行文脈【検索文字列】後続文脈　　（前後の文脈を両方指定）

文脈の条件の必要性については説明が必要であろう。例えば、「のだ」の用例を「[^も][のん](だ|です)」という正規表現を使って検索することを考える。「[^も]」は「ものだ」「もんです」などの用例を除外するために添えているわけであるが、検索文字列に「[^も][のん](だ|です)」という正規表現を指定して検索したのでは、得られる KWIC 索引の１つ――先行文脈に基づくソートによるもの――は次のようになってしまう。

んだ後で、それがはたして自分に何の関係があ【るのだ】ろうと思った。けれども冷やかな無関心の傍にいて送ったんだな」「ええ、文章は浜田が書い【たんです】。僕が名前を借して遠藤が夜あすこのうちまいから、口先で偉そうな事を云って他をごまか【すんだ】ろうと思った。「仕事ができなくって、ただ理いか」とまで云った。彼は「うん、実は行きた【いのだ】が……」と渋っていた。実際これは彼の平生にす。そうして慰めてやると、かえって皮肉を云【うのです】。何だか近来はますます変になるようです」、無意識ながら分別していたらしい。　さあ行【くんだ】と催促された時は、なるほど旅順に来る以上、情だな。僕と戦うんじゃないぜ」「じゃ誰と戦【うんだ】」「君は今すでに腹の中で戦いつつあるんだ。しちゃ毒だ」「腹なんかどうでもいいさ」「痛【むんだ】ろう」「痛む事は痛むさ」「だから、ともかく

これでは「のだ」の索引としていかにも見づらいし、まともな意味での先行文脈に基づくソートになってもいない。要は、求める文字列の先行文脈にすぎない「[^も]」を検索文字列の一部として扱ったところに問題があるわけである。そこで、正しくは、「[^も]」を先行文脈として――すなわち、「[^も]【[のん](だ|です)】」と指定して――検索する。そうすれば次のような KWIC 索引が得られることになる。

か」とまで云った。彼は「うん、実は行きたい【のだ】が……」と渋っていた。実際これは彼の平生にも。そうして慰めてやると、かえって皮肉を云う【のです】。何だか近来はますます変になるようです」「だな。僕と戦うんじゃないぜ」「じゃ誰と戦う【んだ】」「君は今すでに腹の中で戦いつつあるんだ。そ無意識ながら分別していたらしい。　さあ行く【んだ】と催促された時は、なるほど旅順に来る以上、催から、口先で偉そうな事を云って他をごまかす【んだ】ろうと思った。「仕事ができなくって、ただ理窟て送ったんだな」「ええ、文章は浜田が書いた【んです】。僕が名前を借して遠藤が夜あすこのうちまでちゃ毒だ」「腹なんかどうでもいいさ」「痛む【んだ】ろう」「痛む事は痛むさ」「だから、ともかくもだ後で、それがはたして自分に何の関係がある【のだ】ろうと思った。けれども冷やかな無関心の傍に起

同様に、「なかなか」の用例のうち否定の述語が後に続く「なかなか～ない」という形のものだけ――ただし、「～」の部分は 20 文字以内とする――は「なかなか.{1,20}ない」という正規表現によって検索することができるが、「【なかなか】.{1,20}ない」のように指定することにより検索結果で「なかなか」の部分だけをハイライトすることができる。

簡易検索用の定義ファイル KWIC.def に記述された検索行は無視される。

(g) 終了行

連続検索時に定義ファイルに終了行が現れると、KWIC はそこで処理を終了する。それ以後に何が書かれていても処理に影響しない。

終了行は、「終了」と書く代わりに、「end」または「quit」と書くこともできる。

簡易検索用の定義ファイル KWIC.def に記述された終了行は無視される。

6　擬似正規表現

KWIC での検索文字列や前後文脈の指定には、正規表現の代わりに擬似正規表現を使うこともできる。擬似正規表現は、漢字１文字とかマ行五段活用動詞の語尾１文字といった検索条件を簡単に指定するためのものである。

定義されている擬似正規表現は以下の表の通りである。それぞれの擬似正規表現は検索時にその右の値の欄にある正規表現に展開される。「＄」は、一般の正規表現の記号と異なり、全角文字で指定する。

擬似正規表現	値	意味
＄平	[ぁ-ん]	平仮名１文字
＄片	[ァ-ヴー]	片仮名１文字
＄仮	[ぁ-んァ-ヴー]	仮名１文字
＄漢	[亜-熙]	漢字１文字
＄字	[ぁ-んァ-ヴー亜-熙]	仮名・漢字１文字
＄く	[かきくけこい]	カ行五段活用動詞の語尾１文字
＄ぐ	[がぎぐげごい]	ガ行五段活用動詞の語尾１文字
＄す	[さしすせそ]	サ行五段活用動詞の語尾１文字
＄つ	[たちつてとっ]	タ行五段活用動詞の語尾１文字
＄ぬ	[な-のん]	ナ行五段活用動詞の語尾１文字
＄ぶ	[ばびぶべぼん]	バ行五段活用動詞の語尾１文字
＄む	[ま-もん]	マ行五段活用動詞の語尾１文字
＄る	[ら-ろっ]	ラ行五段活用動詞の語尾１文字
＄う	[わいうえおっ]	ワ行五段活用動詞の語尾１文字
＄末	[うくぐすつぬぶむるいなただ]	述語終止・連体形末尾の１文字
＄た	[ただ]	「た」「だ」のいずれか１文字
＄て	[てで]	「て」「で」のいずれか１文字
＄文	[^。！？」]	句点類を除く文字１文字
＄句	[。！？」]	句点類の１文字

例えば、検索文字列に「＄漢+の＄漢+」を指定すれば、実際には「[亜-熙]+の[亜-熙]+」という正規表現に展開して検索される。「探す」「見つかる」の全活用形を表す「探[さしすせそ]」「見つ

か[ら-ろっ]」は「探＄す」「見つか＄る」と指定して検索することができる。「＄末かわりに」は「かわりに」の用例のうち直前が述語であるものだけを検索し、「〜のかわりに」や単独の「かわりに」を除外する。「＄てしま＄う」と指定すれば、「〜てしまう」「〜でしまう」の全活用形を検索することができる。「なかなか＄文+ない」という指定は、「なかなか〜ない」という文字連続のうち、「なかなか」と「ない」が同一文に共起したものに限る――両者のあいだに句点が介在しない――という条件を意味する。「である＄句」は「である」のうち文末の言い切りに使われたものだけを検索し、「〜であるから」「〜である場合」などを除外する。

擬似正規表現が「[...]」「[^...]」の中に置かれたときは、上の表に示した値からその両端の各括弧が取り除いたものに展開される。これにより、擬似正規表現の定める文字の範囲を拡張することができる。例えば、平仮名または踊り字「ゝ」「ゞ」の１文字は「[＄平ゝゞ]」、漢字または踊り字「々」の１文字は「[＄漢々]」で表すことができ、それぞれ「[ぁ-んゝゞ]」「[亜-熙々]」と展開される。漢字と「々」以外の１文字は「[^＄漢々]」と書けば「[^亜-熙々]」と展開される。また、「おっしゃ[＄るい]」とすれば「おっしゃる」の全活用形を検索でき、「[＄末き]」を使えば、「良き」「〜べき」のような「き」で終わる述語も検索の対象に含めることができる。なお、「＄仮」は「[＄平＄片]」、「＄字」は「[＄仮＄漢]」とそれぞれ等価である。

ただし、句点類を除く文字１文字を表す「＄文」だけは例外的に「[...]」「[^...]」の中に置くことができない。

7　補足・注意事項

7.1　KWICが検索対象とする日本語テキストに関わる事項

・処理可能なファイルの種類

KWIC が検索対象として処理できるのは Shift_JIS コードの日本語テキストである。

Shift_JIS コードの日本語テキストでないと容易に推定されるファイルは検索対象としない。

・処理可能なデータの量

検索対象とするテキストの数と分量に事実上制限はない。

一回の検索で処理できる用例数の上限はパソコンや行長の設定などに依存する。筆者の環境では「行長＝50」の設定で用例数 20 万件程度の検索は問題なく処理できた（2007 年に確認）。用例数が限度を超えれば KWIC は異常終了する。目安として、数十万、数百万の用例の検索には KWIC は使えないと考えていただけばよいだろう。

・空行を含まない巨大なテキストの問題

KWIC は、空行（＝文字のない、改行だけの行）を境界とする一まとまりのテキストを単位として検索を行う。その単位をかりに「段落」と呼ぶことにすると、空行を含まない大きなテキストの場合にはテキスト全体が１つの長大な段落ということになってしまう。ところが、それでは KWIC の処理上都合が悪いので、空行なしにテキストが延々と続く場合は便宜上約十万字ごとに行末（改行の位置）で区切り、そこまでを１つの段落と見なすようにしている。

ただ、その扱いによってたまたま検索文字列が前後の段落に分断されてしまい、検索されるべ

き用例が検索から漏れてしまう可能性がある。そのような確率は非常に低いはずであるが、空行を含まない巨大なテキストにはせめて十万字ごとに適宜空行を入れて段落を分けておくのが一番確実ではある。

※Macintosh で作られたテキストを処理する場合に注意を要する点がある。Windows と Macintosh とでは改行コードが異なる関係で、Macintosh で作られたテキストを KWIC で検索するときは、あらかじめテキスト中の改行（＝0dh）を Windows 式の改行（＝0dh＋0ah）に変換しておく必要がある。そうしないと、Macintosh 式の改行を認識できない KWIC にとってはファイル全体が単一の長い行に見えてしまい、そのサイズがあまりに大きければ読み込みに失敗することになるからである。改行コードの変換は秀丸エディタなどのテキストエディタで行うこともできるし、インターネットで「改行コード 変換」で検索すればほかの方法も容易に見つけることができる。

7.2 KWICの用法や処理内容に関わる事項

・複数の検索の同時実行

複数の検索を同時進行で行うことができる。それには、ある検索を実行しているとき、通常通りに KWIC のアイコンをダブルクリックし、KWIC のウィンドウを新たに開けばよい。（ある検索を実行中に、新しいウィンドウを開くことなく現在のウィンドウの「KWIC 実行」のボタンを押しても機能しない。）

大規模なコーパスから文字列を検索するのには長い時間がかかる。そのようなとき、１つの検索が終わるのを待つことなく、簡単に別の検索を始めることができる。複数の検索を同時に行うと個々の検索の速度は落ちるが、それでも１つずつ順番に作業を進めるのに比べれば全体として効率がよい。

ただ、注意を要するのは、複数の検索による結果出力で競合が生じないよう、各検索において相異なる出力ファイル名を指定する必要があることである。簡易検索の場合も、連番の付加に頼らず、出力ファイル名を明示的に指定するようにする必要がある（複数の検索がほぼ同時に出力を始めた場合、同一の連番が使われて競合が生じる可能性がある）。

・複数コーパスの使い分け

複数のコーパス、例えば、小説、雑誌、新聞Ａ、新聞Ｂという４種類のコーパスがあり、それぞれが単一ないし複数のファイルとして C:¥corpora¥novels、C:¥corpora¥magazines、C:¥corpora¥newpaper_a、C:¥corpora¥newspaper_b というディレクトリに収められているものとする。

※各コーパスが C:¥novels、C:¥magazines、C:¥newpaper_a、C:¥newspaper_b という相互に独立したディレクトリに入っているということでもよいし、コーパスがすべて文学作品で、それが作家、ジャンル、時代などによって複数のディレクトリに分けて収めてあるということでももちろんよい。
※ディレクトリ名は、小説、雑誌、新聞Ａ、新聞Ｂ のように全角文字を含んでいてもかまわない。

それらの複数のコーパスを使い分けるための方法は簡易検索と連続検索とで異なる。

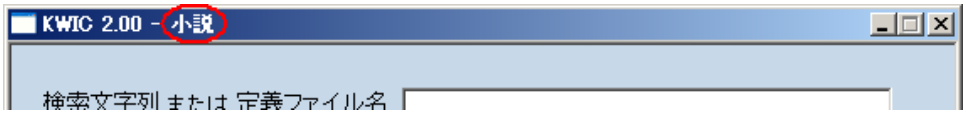
連続検索の場合は話が単純で、定義ファイルにおいて必要に応じてコーパスの所在を「パス＝～」で指定すればよい（定義ファイルのサンプル sample4.def を参照）。

他方、簡易検索で複数コーパスを使い分けるには、KWIC のファイルをコーパスごとに異なる

ディレクトリにコピーして使う。具体的には以下の準備を行う。

(1) C:¥KWIC の中に、各コーパスに応じた novels、magazines、newspaper_a、newspaper_b というサブディレクトリを作る。
(2) C:¥KWIC にある KWIC という名前の３つのファイルを(1)で作った各サブディレクトリの中にコピーする。
(3) 各サブディレクトリにコピーされた KWIC.def の「パス＝～」の設定を適宜書き換える。例えば、novels の中の KWIC.def については「パス＝C:¥corpora¥novels」とする。
(4) 各サブディレクトリにコピーされた KWIC（HTML Application）のそれぞれについて、デスクトップ上にショートカットアイコンを作る。その際、デスクトップには同名のアイコンを複数置くことができないので、アイコンを１つ作るごとにその名前を適宜変更する。例えば、novels のアイコンは「KWIC novels」「KWIC 小説」「novels」といった名前にする。

なお、複数の KWIC を区別できるように、KWIC の置かれたディレクトリの名前が KWIC 以外であるときは、タイトルバーにそれが表示されるようにしてある。例えば、C:¥KWIC¥小説 という名前のディレクトリに置かれた KWIC の場合は次のようになる。



・エクセルファイルの列幅の調整

生成されるエクセルファイルの列幅は自動的に設定されるが、環境によって検索文字列の列幅にわずかな過不足が生じることがある。そのような場合、列幅の拡大・縮小率（例えば 1.15）を書いたファイルを作って C:¥Windows¥Temp¥KWIC.dat という名前で保存しておけば、検索文字列の列幅を補正することができる。指定可能な拡大・縮小率の範囲は 0.8～1.25 である。

・英単語などの検索

簡易検索時に半角文字だけから成る検索文字列を指定するときは、それが連続検索の定義ファイル名でなく簡易検索の検索文字列の指定であることを明示するために、検索文字列の前に半角文字の「!」を添える。例えば、Japan という文字列を検索したいときには、検索文字列を「!Japan」という形で指定する。

・ファイル名に使えない文字

Windows では一般にファイル名に以下の半角文字を含めることができない。定義ファイルと出力ファイルの名前においても同様である。

" * + / : < > ? ¥ |

・出力ファイル名および定義ファイル名のパス指定

簡易検索・連続検索時の出力ファイル名および連続検索時の定義ファイル名の指定にあたっては、必要に応じてドライブ名やディレクトリ名を加えることができる。

例：C:¥用例¥nihongo　（出力ファイル名の指定）
C:¥KWIC¥def¥kensaku.def　（定義ファイル名の指定）
出力＝D:¥用例¥検索結果　（定義ファイルでの出力ファイル名の指定）

出力ファイルを指定のディレクトリに作成する場合、そのディレクトリはあらかじめ作成しておく必要がある。

・使用可能な正規表現の要素と指定された正規表現のチェック

KWIC では一般的な正規表現の要素をすべて使うことができる。ただし、検索文字列の前後の文脈を指定するときには後方照応（¥1、¥2、～）は使えない。

不正な正規表現を指定した場合はエラーメッセージを出して処理を中止する。ただし、正規表現の誤りのチェックは十分に厳格なものではなく、不正な正規表現でも（英語の警告メッセージが表示されて）処理が続行される場合がある。

・簡易検索時の出力ファイル名に加えられる連番の決定方法

簡易検索時に出力ファイル名に３桁の連番が加えられるときは、ディスク上にすでに存在する同名ファイルのうち接尾辞「_0」の付いたものの連番の最大値に１を加えたものが使われる。例えば、出力ファイル名が name.txt で、ディスク上に name001_0.txt、name002_0.txt、name005_0.txt が存在する場合、連番として（003 でなく）006 が選ばれる。ただし、すでに連番 999、接尾辞「_0」の同名ファイルが存在する場合は、再び連番 999 が使われる。

このように接尾辞「_0」のファイルは検索終了後も意味を持つので、接尾辞「_0」のファイルだけを消去することは避けるほうがよい。接尾辞「_0」のファイルは、接尾辞「_1」～「_3」のファイル（や同名のエクセルファイル）を消すときに同時に消去するようにするのがよい。

・ファイルの上書き

出力ファイル名に連番が付加されない場合――すなわち、連続検索時と、簡易検索で出力ファイル名をコマンドラインで指定したとき――、および、簡易検索で連番 999 のファイルに出力する場合などは、出力ファイルと同名のファイルがすでにディスク上に存在していれば KWIC の実行によって上書きされる。

ただし、出力ファイル名と同じ名前のエクセルファイルがすでに開かれているときはファイルを保存することができない。その場合必要とあらば、開いているファイルを閉じるか、もしくは、別名を指定することによって保存する。

・ファイルの消去

連番が付加されるかどうかにかかわらず、出力ファイルと同名のファイルがディスク上に存在

する状態で検索を行った場合、その検索によって出力されない接尾辞のファイルは消去される。例えば、name_0.txt～name_3.txt のファイルがディスク上に存在する状態で出力ファイル名 name.txt を指定して検索を行い、その結果として name_0.txt～name_2.txt の３つのファイルが出力されるときには name_3.txt は消去される。

・タグの出力幅

前置形式でタグを付加するときは、タグの出力幅は当該の検索結果の中で最長のタグの長さに統一される。タグの出力幅を一定にしなければ KWIC 索引がきわめて見づらいものになるので、それを避けるための措置である。

・行連結によって生じ得る問題

日本語テキストが桁折りされていて語句が前後の行に分断されている場合でもそのまま検索できるようにするために、KWIC では隣り合う行を連結して検索する。この処置がもたらす問題は、表現としてつながっていないときにも行が連結され、無用の検索結果が得られる可能性があるということである。

例えば、次のような行単位での列挙においてそうした問題が生じ得る。

おーい、中村君
月がとっても青いから
上を向いて歩こう
だからいったじゃないの
もしも月給が上がったら
ハイそれまでヨ
達者でな
お富さん

この列挙の行頭にも行末にも空白が付いていないとすれば、行の連結によって「おーい、中村君月がとっても青いから上を向いて歩こうだからいったじゃないのもしも月給が上がったらハイそれまでヨ達者でなお富さん」という長い“文”ができてしまう。その結果、例えば「～から～しよう」というパターンの表現の検索にかかってしまうことになる。もっとも、このように行連結のために無用の検索結果が得られる確率は低いであろうし、行を連結したために検索には付き物のゴミが検索結果にわずかに混じるのと行を連結しないために必要な検索結果が不完全にしか得られないのとでは前者のほうがはるかに問題が軽いと筆者は判断する。

・タブコードの扱い

日本語テキストに含まれるタブコードは半角文字の空白に置換して処理される。

タブコードによって KWIC の見かけ上の行長が不揃いになるのを防ぐためである。

・検索の一時停止

画面上で検索結果が流れて行くのを一時的に止めて様子を見たいときは、コマンドプロンプトをアクティブウィンドウにした状態で[Ctrl]+ [S]を押せばよい。再び[Ctrl]+ [S]またはほかのキーを押せば処理が再開する。

・処理の中止

進行中の KWIC の処理を中止するための“穏やか”な方法はない。処理を打ち切るには、コマンドプロンプトのウィンドウの右上隅の[×]ボタンをクリックして、コマンドプロンプトを終了するのが簡単である。ほかに、コマンドプロンプトをアクティブウィンドウにした状態で[Ctrl]+ [C]を押して KWIC の実行を強制的に終了させるという方法もある。

・正規表現指定上の注意その他

検索文字列の指定に用いた正規表現に極端に長い文字列が該当（マッチ）すると KWIC は異常終了する（しかるべきメッセージを表示することなく実行が終了する）。コマンドプロンプトで KWIC を実行して（補説 C を参照）同じ検索を行えば、画面上に「Stack overflow in regexp matcher」という Ruby のエラーメッセージが表示されることから確認できる。そのような場合は、正規表現に極端に長い文字列が該当しないよう条件を調整して検索し直す必要がある。

その他の特殊な条件下でも KWIC が異常終了することは考えられる。そのようなときはコマンドプロンプトで同じ検索を行えば、Ruby の出すエラーメッセージを見ることができる。ただし、原因が Ruby 処理系にある問題は根本的な対処がむずかしい可能性が高い。

補説A　検索文字列の広い文脈を参照する方法

1　概要

用例閲覧時に検索文字列の現れる文脈をより広く参照したいことがある。エクセルファイルとテキストファイルに分けてその方法を説明する。

2　エクセルファイルの場合

2.1　表示幅の変更

検索結果を収めたエクセルファイルでは、それぞれの列はあらかじめ設けられた基準に基づく幅に設定される。検索文字列の前後の文脈は十数文字ずつ表示される。

より広い範囲の文脈を参照するには列の幅を広げればよい。あるいは、特定のセルの内容を確認するだけなら、セルをクリックしてその内容をウィンドウ上方の数式バー（fx という表示の右側の欄）に表示させるという方法も可能である。それらの方法で参照できる文脈の範囲が狭すぎる場合は、定義ファイルにおける行長の設定を大きくして検索すればよい。

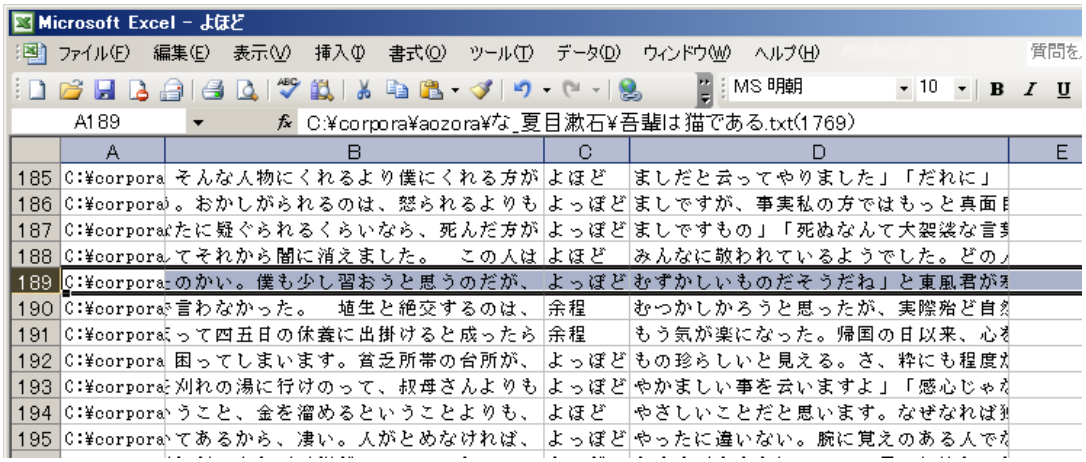
2.2　原文テキストの参照

次のページにある拙作ソフト STRetriever を使えば、当該の用例を含む原文テキスト全体の当該箇所を簡単に参照することができる。

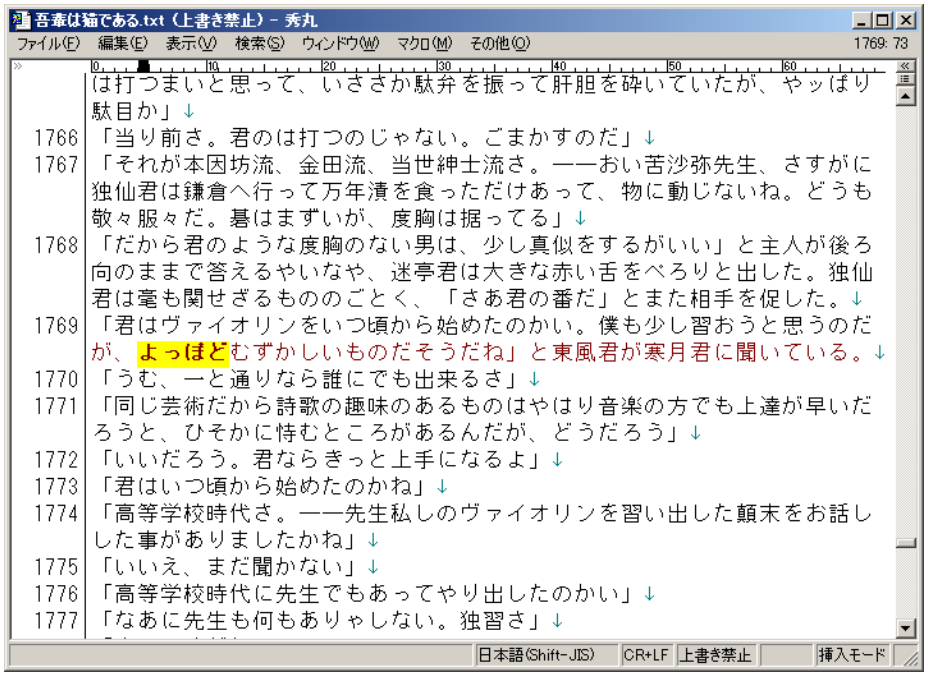
http://www.tanomura.com/research/STRetriever/

※Window の更新に付いていけなくなり、このソフトの更新と公開は現在停止しています。

例えば、次のような検索結果において、



特定の用例を選んでコピーして STRetriever のアイコンをダブルクリックすると、次のように原文テキストの当該箇所が秀丸エディタで開かれ、検索文字列はハイライト表示される。

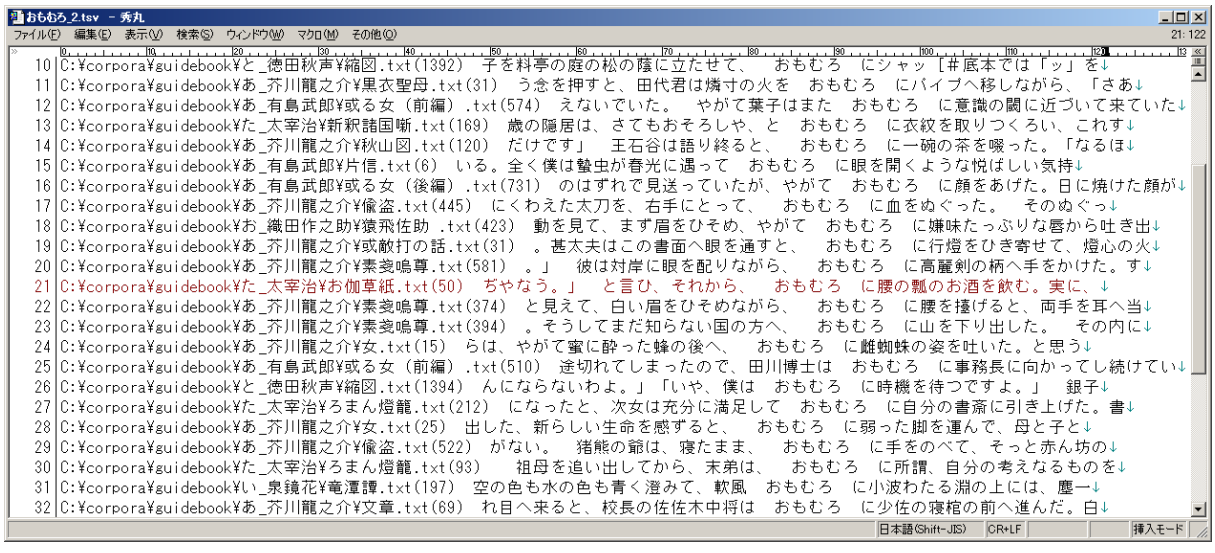


ただし、条件によってはハイライト表示がなされない場合がある（詳しくは STRetriever のページを参照)。

3　テキストファイルの場合

STRetriever の使えない環境（簡体中文版の Windows など）では、やや不便であるが、検索結果をタグ前置の形式で出力し、テキストエディタのタグジャンプの機能を用いればよい。

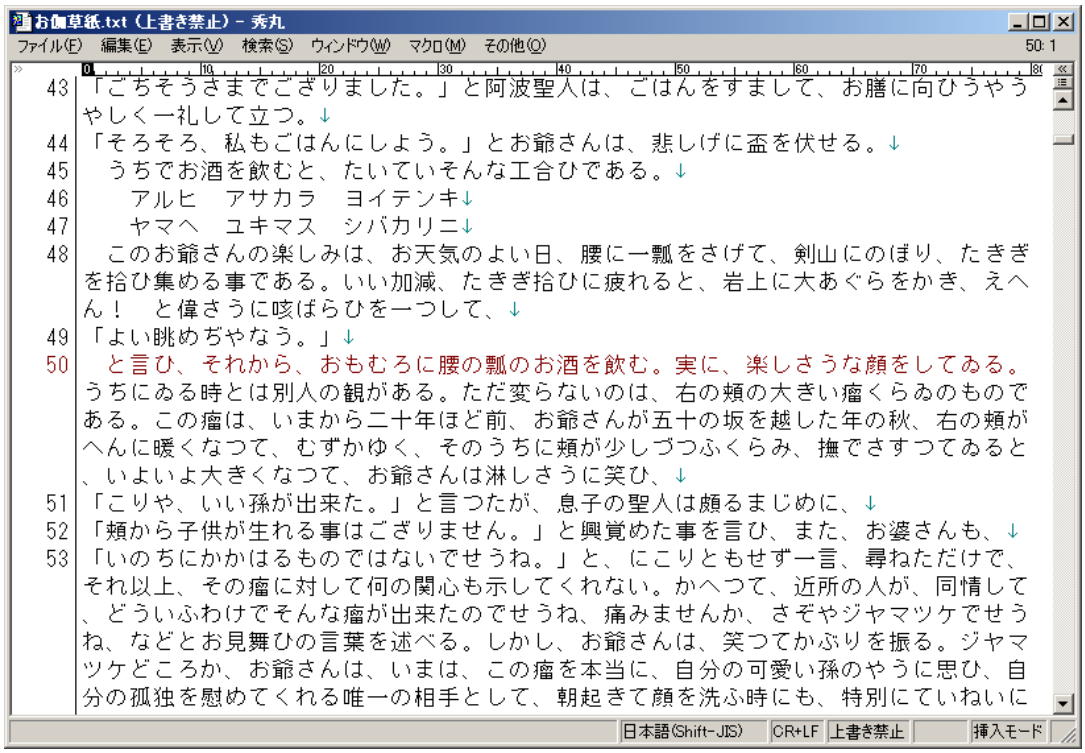
例えば、次の図は「おもむろ」を検索して出力された おもむろ_2.tsv を秀丸エディタで開いたところである。



秀丸エディタでタグジャンプを実行するには、注目している用例の行の上にカーソルを置いた

状態で、メニューの［その他］から［タグジャンプ］とするか、もしくは、[F10]を押せばよい。これにより原文テキストが開かれ、当該箇所が表示される。

上の画面の状態（着色表示の行にカーソルが置かれた状態）でタグジャンプを実行すると次の図のようになる。



ただし、この方法では検索文字列はハイライト表示されない。また、カーソルは当該行の冒頭に置かれるので、１行が画面上の複数行に渡る場合は検索文字列は画面上の２行目以後にある可能性がある。

タグジャンプの機能は主要なテキストエディタにはたいてい備わっている。

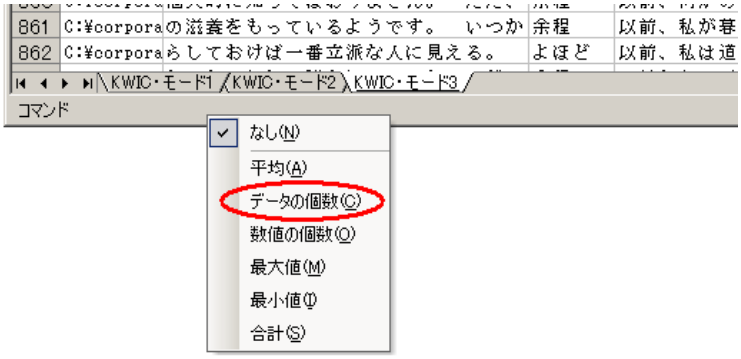
補説B　用例数を簡単に知る方法

1　概要

エクセルの機能を利用すれば、ソートされた KWIC 索引の特定の範囲に含まれる用例の件数を簡単に知ることができる。

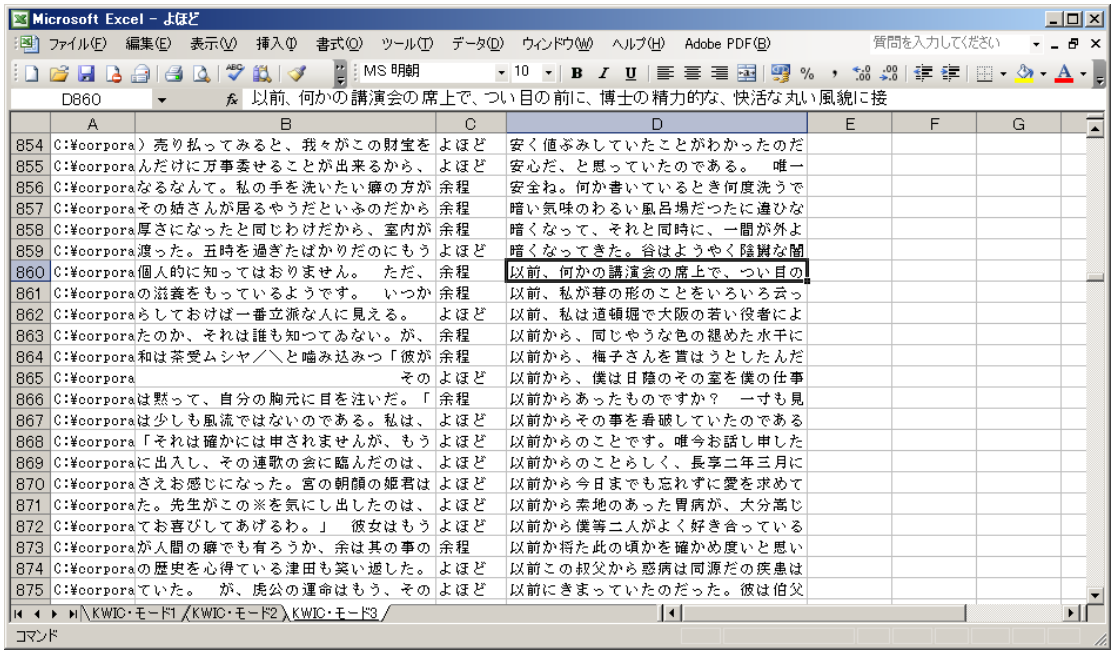
2　準備

エクセルのウィンドウ下端のステータスバー（左端に「コマンド」または「準備完了」などと表示されている部分）を右クリックし、「データの個数」にチェックが入っていなければ左クリックする。すでにチェックが入っていれば何もする必要はない。

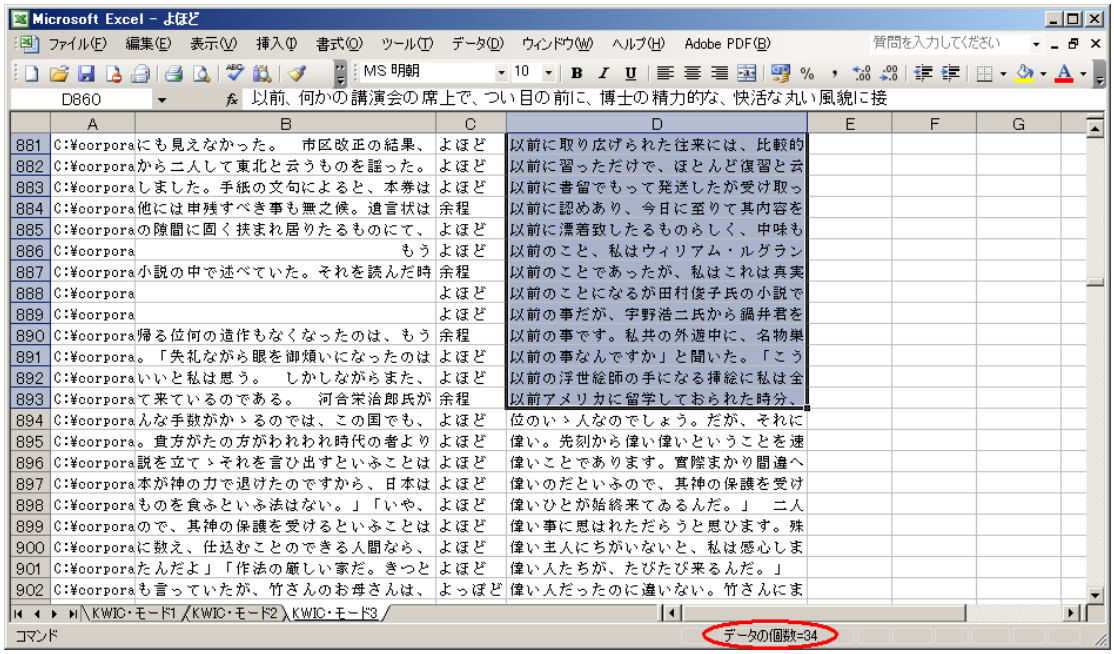


3　用例数の調べ方

例えば次の「よほど」の KWIC 索引で「よほど」に「以前」が続く用例の件数を知るには、まずその最初の用例（の任意の列のセル）にカーソルを合わせて左クリックする。これにより、セルの枠が太線に変わる。



次に KWIC 索引を必要に応じてスクロールし、最後の用例（の同じ列のセル）にカーソルを合わせて、今度は[Shift]キーを押し下げた状態で左クリックする。これにより次のように範囲が選択された状態になり、同時に、選択されたセルの個数がステータスバーに「データの個数＝～」という形で表示される。



※データの個数は選択された空でないセルの個数を示す。範囲の指定には空のセルを含まないどの列を選んでもかまわない。
※選択したセルの文字色や背景色を変更すれば、数え終えた用例を一目で区別できるようになる。用例の種類ごとに色を変えれば分類を示すこともできる。
※[Ctrl]キーを押し下げた状態でセルを左クリックすれば不連続のセルを選択することができる。また、[Ctrl]キーとマウスの左ボタンを両方押し下げた状態でマウスを動かすことによってセルを連続的に選択することができる。

補説C　コマンドプロンプトでのKWICの実行

1　概要

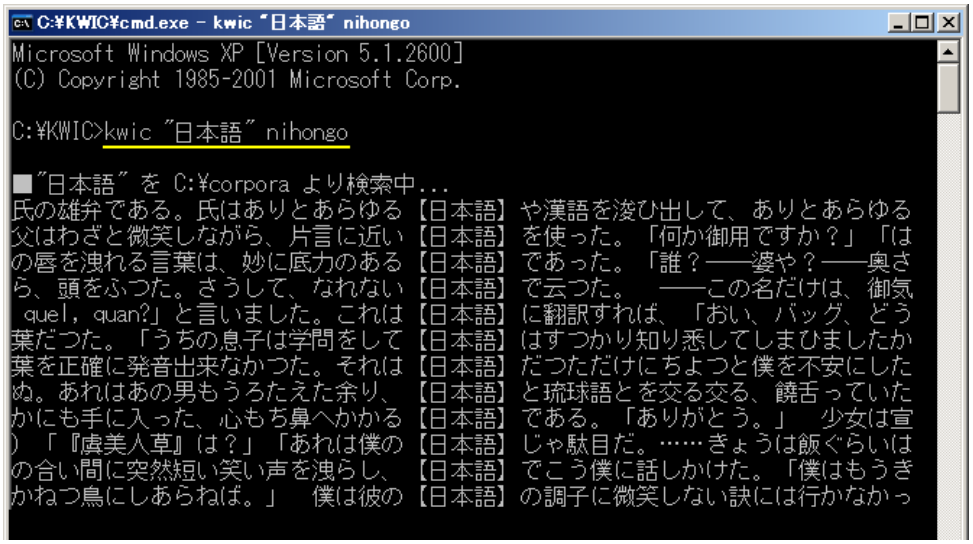
KWIC は本来コマンドプロンプトで実行するソフトウェアである。コマンドプロンプトで実行すれば、コマンドプロンプト上で行えるさまざまな処理と組み合わせて KWIC を使用することができる。

※コマンドプロンプトの開き方およびそれを便利に使うためのヒントについては、「日本語 KWIC 索引生成ソフトウェア KWIC」のページに掲載しているコマンドプロンプトの解説を参照されたい。また、コマンドプロンプトでの複雑な作業を劇的に効果的にすることのできるバッチファイルについては荻野綱男・田野村忠温編『Ruby によるテキストデータ処理』（明治書院、2012）の「付録 B バッチファイル」で解説している。

2　コマンドプロンプトでKWICを実行する方法

簡易検索を行うには、コマンドラインで次の下線部のようにタイプして[Enter]キーを押す。出力ファイル名の指定は省略することもできる。引用符「”“」、および、「kwic」に始まる３要素を隔てる空白にはいずれも半角文字を用いる。半角文字入力モードと日本語入力モードの切り替えは[Alt]+[半角/全角]によって行う。

C:¥KWIC>kwic　”検索文字列”　出力ファイル名



連続検索を行うには、コマンドラインで次の下線部のようにタイプして[Enter]キーを押す。

C:¥KWIC>kwic 定義ファイル名

3　補足説明

・引用符の省略

検索文字列がコマンドラインで特別な意味を持つ文字・記号を含まない場合は、検索文字列を

囲む引用符 「”“」は省くことができる。日本語表現の検索という目的から現実的に言えば、引用符が必要となるのは正規表現の指定時に「^」や「|」の記号を使う場合くらいのものである。

しかし、引用符を省く習慣を身に付けると引用符が必要なときに付け忘れて正しく検索できないことになるので、常に引用符を使うようにするのが無難である。

・複数コーパスを使い分ける方法

マニュアル本文の 7.2 で説明した複数コーパスの使い分けを行う場合は、「cd ¥KWIC¥novels」などとしてカレントディレクトリを適宜変更したうえで検索を行う。検索結果は、特に出力ファイルの場所を指定しなければ当該のディレクトリ（C:¥KWIC¥novels など）に出力される。また、KWIC.def での「出力＝～」の設定にドライブ名・ディレクトリ名を加えておけば――例えば、「出力＝C:¥KWIC¥novels」としておけば――、実行時にファイル名を省略した場合、検索結果を指定のディレクトリ（C:¥KWIC）に、しかも、コーパスの種類が分かる形で（novels001_1 のようなファイル名で）出力することができる。

・英単語などの検索

簡易検索時に半角文字だけから成る検索文字列を指定するときは、それが連続検索の定義ファイル名でなく簡易検索の検索文字列の指定であることを明示するために、検索文字列の前に半角文字の「!」を添える。

C:¥KWIC>kwic ！ "Japan" 出力ファイル名

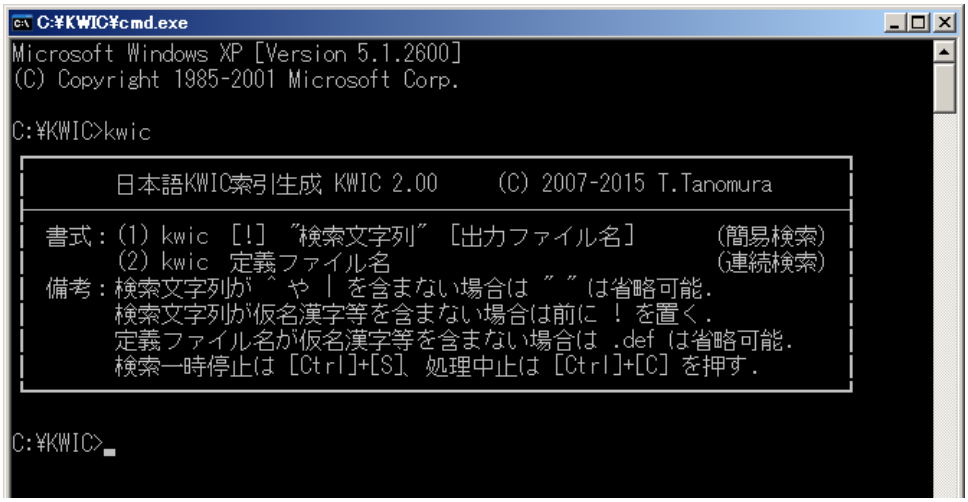
・コマンドライン引数の解釈について

コマンドラインで「kwic ～」とした場合、「～」が検索文字列と解釈されるか、定義ファイル名と解釈されるかは以下の優先順位で決まる。

(1)「kwic ！ ～」のように「!」が前置されていれば「～」は検索文字列と解釈される。
(2) 「～」が.def で終わっていれば「～」は定義ファイル名と解釈される。
(3) 「～」が仮名・漢字または正規表現を表す記号の一部（?*+|(){}[]）を含めば「～」は検索文字列、含まなければ定義ファイル名と解釈される。

・用法表示

コマンドラインで検索文字列や定義ファイル名を指定しないで「kwic」[Enter]とすると、KWIC のバージョンと簡単な用法（コマンドラインでの書式など）が表示される。



付記　日本語KWIC索引生成ソフトウェアKWICの初版は田野村忠温・服部匡・杉本武・石井正彦『コーパス日本語学ガイドブック』（文部科学省科学研究費補助金特定領域研究「日本語コーパス」日本語学班、2007年）の添付CD-ROMに収めて公開した。本文書はそれに改良を加えた最新版のマニュアルである。

本文書の改訂は長期間にわたって散発的に行っていることから、内容に不整合などの問題が生じている可能性が高い。お気付きの点があればご一報いただければ幸いである。