全文検索システム『ひまわり』講習会

全文検索システム『ひまわり』講習会

山口昌也（国立国語研究所／東京外国語大）

本日の内容

- テキストデータにタグ付けを行い,『ひまわり』で活用する方法を紹介
  - 『ひまわり』(ver.1.6.10) + MeCab (ver.0.996)
  - 青空文庫作品+小さいサンプル

- 全体的な流れ
  - 『ひまわり』の紹介と基本的な使い方
  - テキストデータのインポート　+　形態素解析
    (青空文庫作品への自動タグ付け)
  - 『ひまわり』用データの構造と人手アノテーション
    - XMLでのタグ付け
    - 簡易タグでのタグ付け
    - 音声データとの関連付け

ツール・資料などの確認

- 『ひまわり』のインストール
- MeCabのインストール
- テキストエディタのインストール
- 当日配布資料
  - 実習の答えは, 配布データの次のフォルダを参照
    [data]/etc/タグ付け例（実習後参照）

『ひまわり』とは

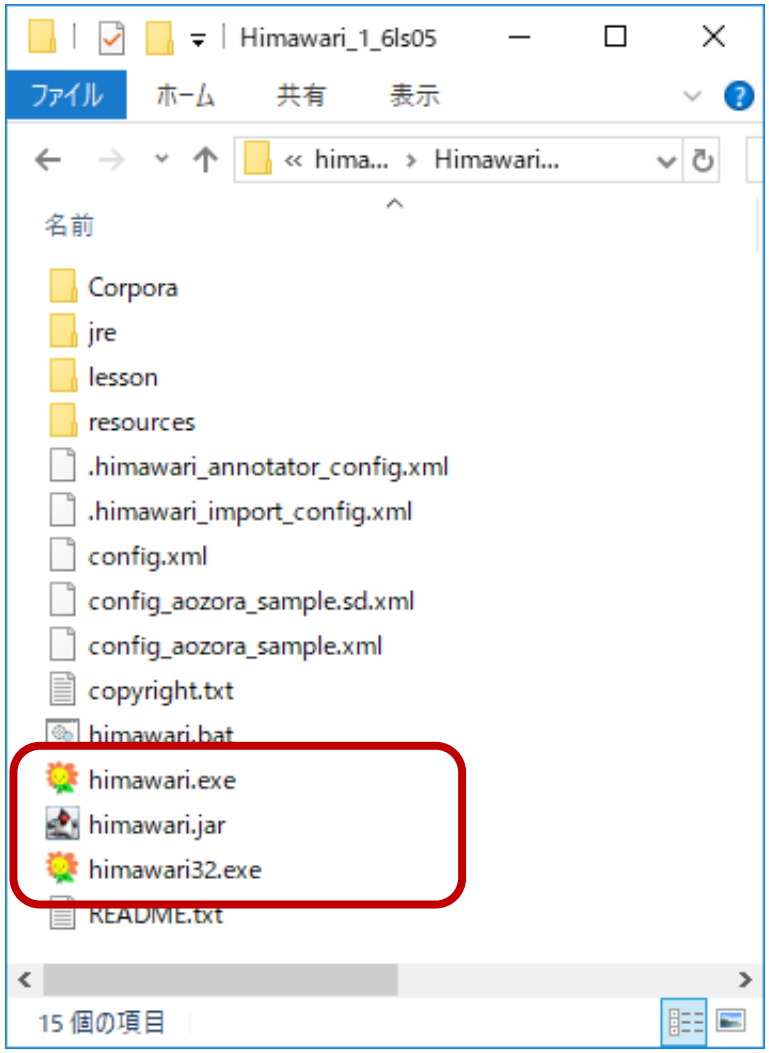
- 言語研究用の全文検索システム
  - 指定された文字列を網羅的に検索して，前後文脈付きで結果を表示します（コンコーダンサ）
  - 『太陽コーパス』（20世紀初頭の総合雑誌『太陽』）用の検索システムとして構築しました

- 特徴
  - XMLでタグづけされたコーパスを全文検索できます
  - Windows, Mac OS, Linux など，多くのOS上で動作します
  - 無料です

Ver.1.6 ⇒　統計的な分析に必要なデータの収集支援機能を強化
（例：総文字数，総単語数）

『ひまわり』の基本的な使い方

『ひまわり』の起動と『ひまわり』フォルダの確認（Windowsの場合）



himawari.exe

普段使うとき
（Windows 専用，64ビット版）
himawari.exe

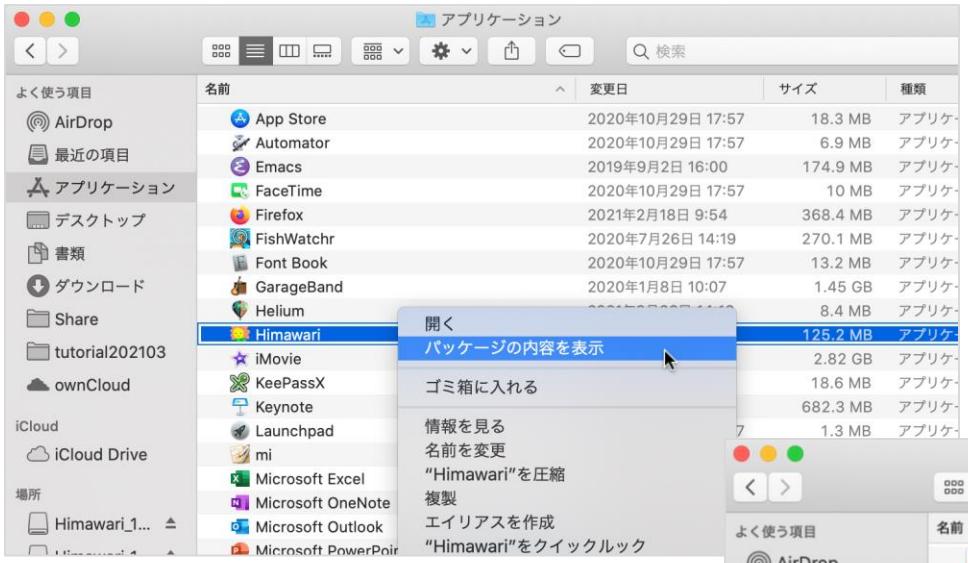
himawari32.exe

himawari.exeが動かないとき
（Windows 専用，32ビット版）
himawari32.exe

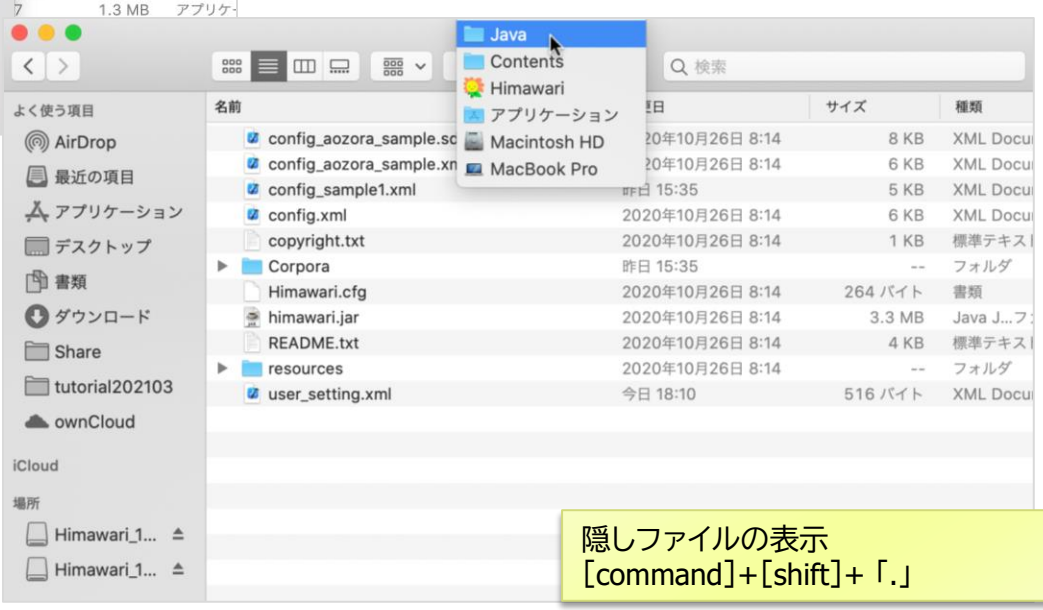
himawari.jar

汎用
（Windows, Mac, Linux など）
himawari.jar

『ひまわり』の起動と『ひまわり』フォルダの確認（macOSの場合）



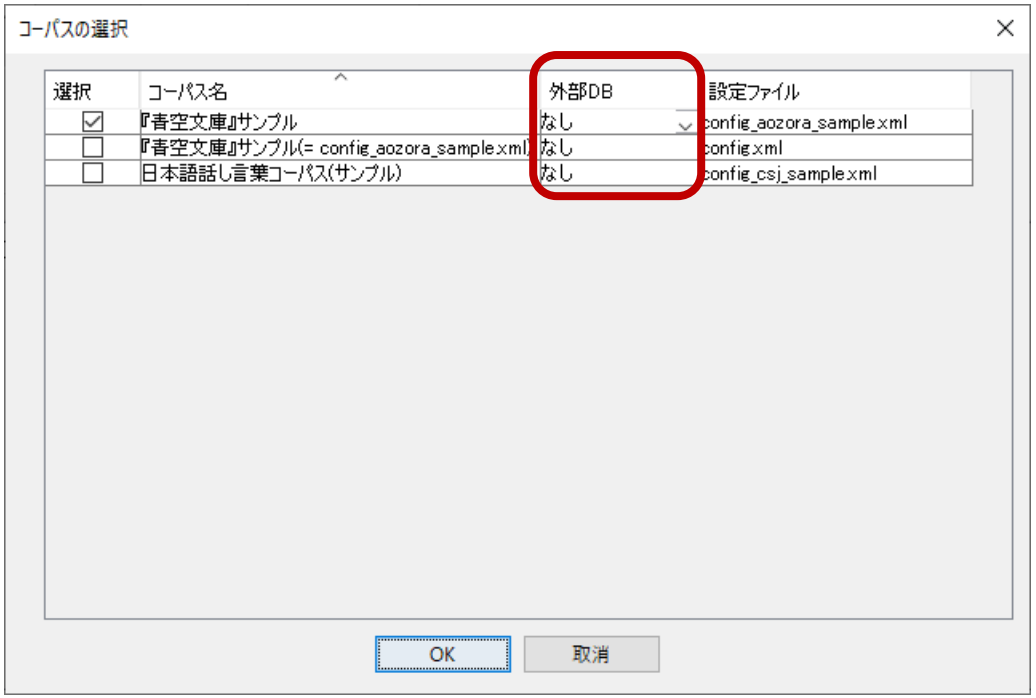
[アプリケーション]→ [Himawari]→ [Contents]→ [Java]



隠しファイルの表示
[command]+[shift]+「.」

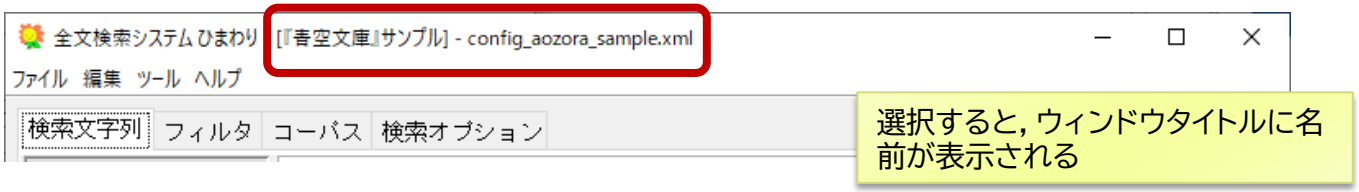
コーパスの選択

- [ファイル]⇒[コーパス選択]



- 「外部DB」
  - コーパスファイルに直接記述していない付与データを格納
  - 『青空文庫』サンプルの場合は，形態素解析結果

- 従来どおり，設定ファイルを『ひまわり』にドロップする方法でもOK



検索する

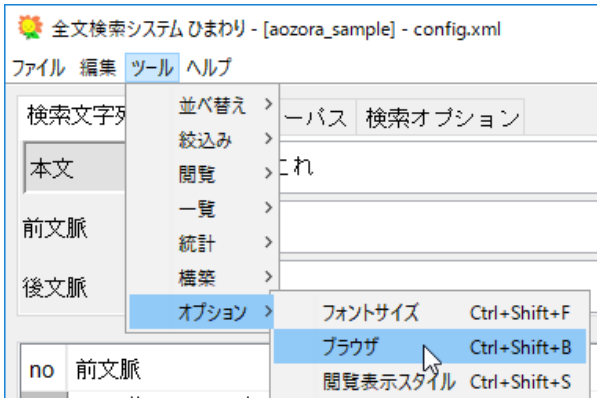

「検索文字列」欄では右クリックで履歴表示
全文検索システム ひまわり - [aozora_sample] - config.xml
ファイル 編集 ツール ヘルプ
検索文字列
検索の実行
検索文字列
フィルタ
コーパス
検索オプショ
本文
これ
前文脈
で終る
後文脈
で始まる
検索
字体変換
クリア
no
前文脈
キー
後文脈
Path
タイトル
著者
検索結果
検索総数:597
検索総数
途中経過の表示

ブラウザでの閲覧

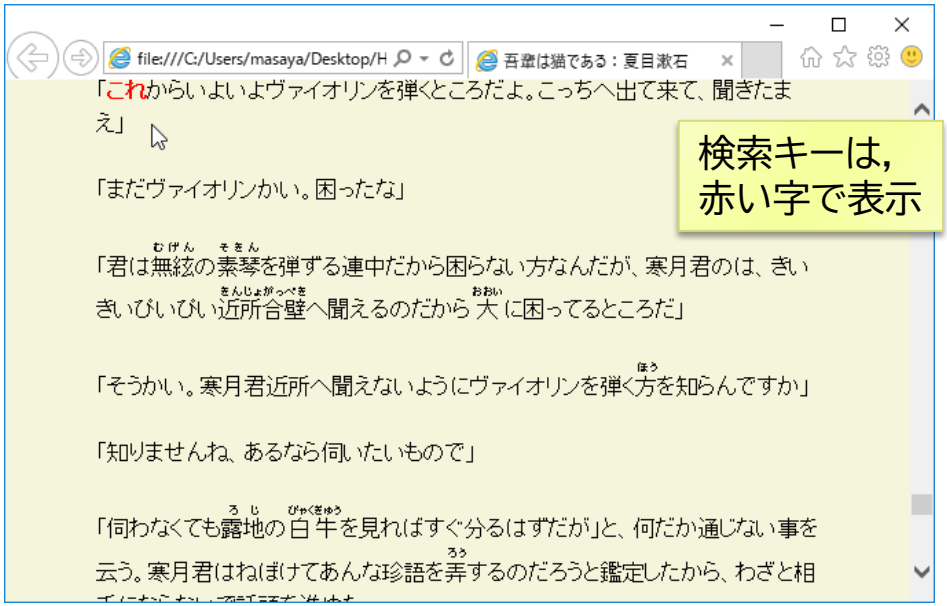
no	前文脈	キー	後文脈	Path
1	指して、しきりにかれ	これ	いいたがるのを、始め	/aozora_s...
2	一軒屋を敲いて、これ	これ	かようかようしかじか	/aozora_s...
3	弾くところです」「	これ	からいよいよヴァイオ	/aozora_s...
4	い話があるかい」「	これ	からいよいよヴァイオ	/aozora_s...
5	、蛸壺峠へかかって、	これ	からいよいよ会津領へ	/aozora_s...
6	見当がつかない」「	これ	からいよいよ弾くとこ	/aozora_s...
7	めちゃなお困ります。	これ	からがいよいよ佳境に	/aozora_s...

閲覧したい用例をダブルクリック

■閲覧用のブラウザの変更



[ツール]⇒[オプション]⇒ [ブラウザ]



検索結果のソート

列名を左クリック

no	前文脈	キー	後文脈	Path	タイトル	著者
1	指して、しきりにかれ	これ	いいたがるのを、始め	/aozora_s...	こころ	夏目漱石
2	一軒屋を敲いて、これ	これ	かようかようしかじか	/aozora_s...	吾輩は猫...	夏目漱石
3	弾くところです」「	これ	からいよいよヴァイオ	/aozora_s...	吾輩は猫...	夏目漱石
4	い話があるかい」「	これ	からいよいよヴァイオ	/aozora_s...	吾輩は猫...	夏目漱石
5	、蛸壺峠へかかって、	これ	からいよいよ会津領へ	/aozora_s...	吾輩は猫...	夏目漱石
6	見当がつかない」「	これ	からいよいよ弾くとこ	/aozora_s...	吾輩は猫...	夏目漱石
7	めちゃなお困ります。	これ	からがいよいよ佳境に	/aozora_s...	吾輩は猫...	夏目漱石
8	うと云うんです。さあ	これ	からがいよいよ失恋に	/aozora_s...	吾輩は猫...	夏目漱石
9	はすこぶる不憫だよ。	これ	からがいよいよ巧妙な	/aozora_s...	吾輩は猫...	夏目漱石
10	充分あらわれている。	これ	からが化物の記述だ。	/aozora_s...	吾輩は猫...	夏目漱石
11	か両君能く聞き給え、	これ	からが結論だぜ。—	/aozora_s...	吾輩は猫...	夏目漱石

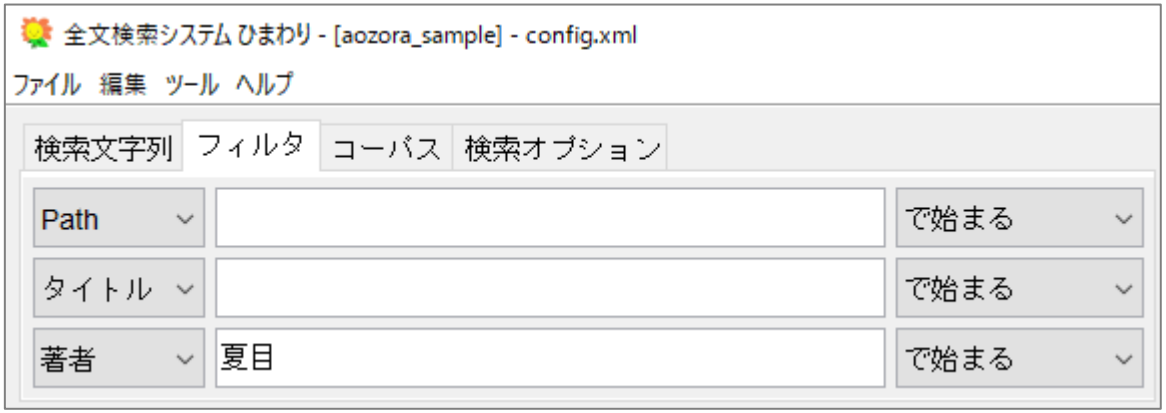
- 昇順
  - 列タイトルをクリック
- 降順
  - シフトキーを押しながら
    列タイトルをクリック

- 複数列を考慮したい場合
  - 優先順位の逆順でソートを実行

例:「話者」ごとに「後文脈」でソート
→ 「後文脈」「話者」の順

検索結果の絞り込み

- 検索時に指定



「著者」欄が「夏目」で始まる結果のみに絞り込まれる

- 検索後に絞り込み

列名を右クリック

no	前文脈	キー	後文脈	Path	タイトル	著者
1	指して、しきりにかれ	これ	いいたがるのを、始め	/aozora_s...	こころ	夏目
	これ	これ	かようかようしかじか	/aozora_s...	吾輩は猫...	夏目
	「	これ	からいよいよヴァイオ	/aozora_s...	吾輩は猫...	夏目
	「	これ	からいよいよヴァイオ	/aozora_s...	吾輩は猫...	夏目
	て、	これ	からいよいよ会津領へ	/aozora_s...	吾輩は猫...	夏目漱石
	「	これ	からいよいよ弾くとこ	/aozora_s...	吾輩は猫...	夏目漱石
	す。	これ	からがいよいよ佳境に	/aozora_s...	吾輩は猫...	夏目漱石

[文字列指定]
[置換]
夏目漱石
芥川龍之介

絞り込みたい値を選択
⇒右クリック
⇒フィルタでもOK

検索結果の頻度集計

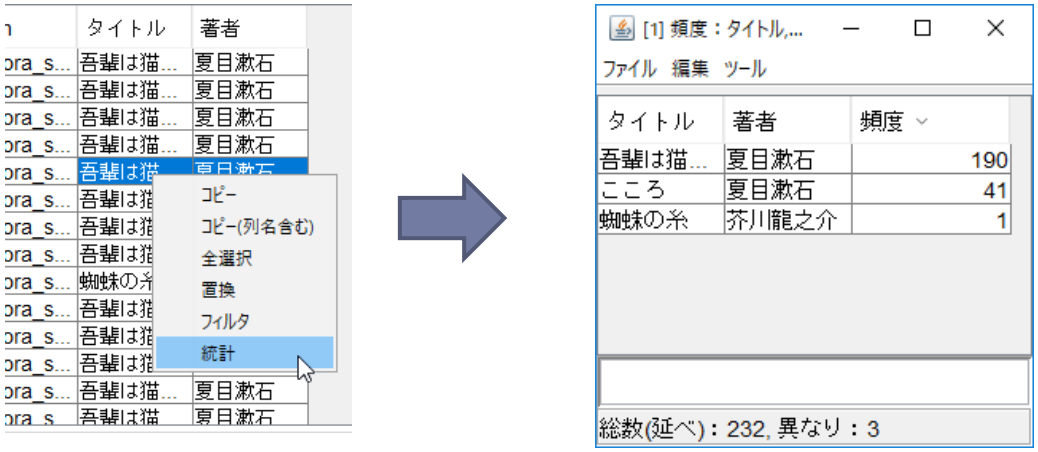
1. 集計したい列を選択

no	前文脈	キー	後文脈	Path	タイトル	著者
1	これは本当の嘶だと、	あの	うそつきの爺やが申し	/aozora_s...	吾輩は猫...	夏目漱石
2	ました、なに猫だから	あの	くらいで充分浄土へ行	/aozora_s...	吾輩は猫...	夏目漱石
3	が来ましたぜ。月並も	あの	くらいになるとなかな	/aozora_s...	吾輩は猫...	夏目漱石
4	まで随分ひきましたが	あの	くらい美しい音が出た	/aozora_s...	吾輩は猫...	夏目漱石
5	なら、立町は豚仙さ、	あの	くらい食い意地のきた	/aozora_s...	吾輩は猫...	夏目漱石
6	ますまい」と云う。「	あの	ちょっとくらい外出致	/aozora_s...	吾輩は猫...	夏目漱石
7	雪江さんが聞く。「	あの	ね。あとでおならは御	/aozora_s...	吾輩は猫...	夏目漱石
8	さんは謙遜した。「	あの	ね。坊たん、坊たん、	/aozora_s...	吾輩は猫...	夏目漱石

複数の列を
選択することも可

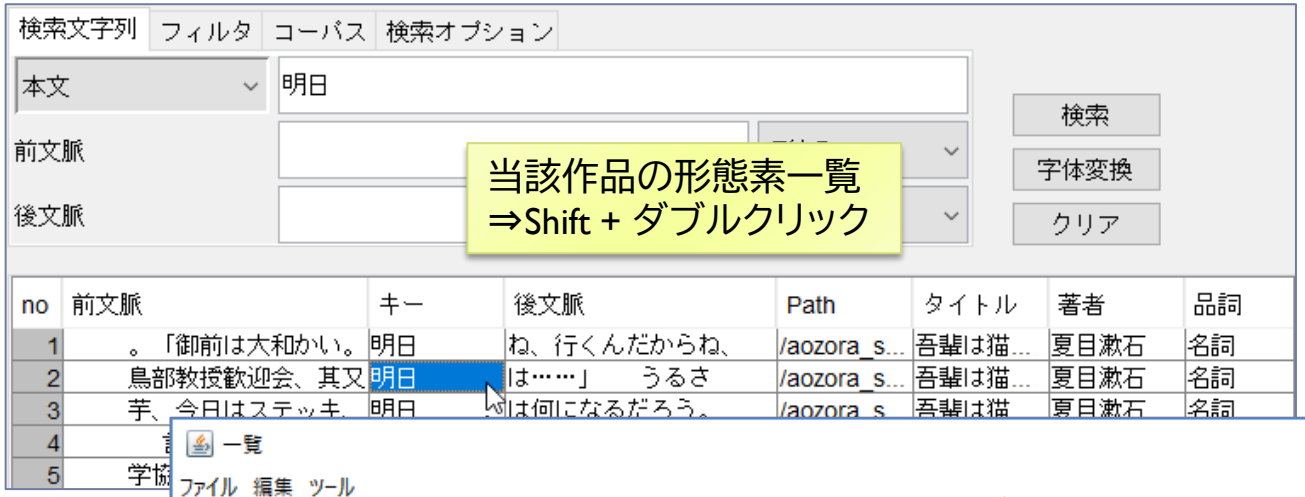
離れた列の選択
- WindowsはCtrlキー
- macOSはcommandキー

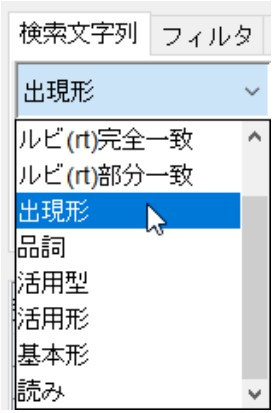
2. 右クリック⇒「統計」

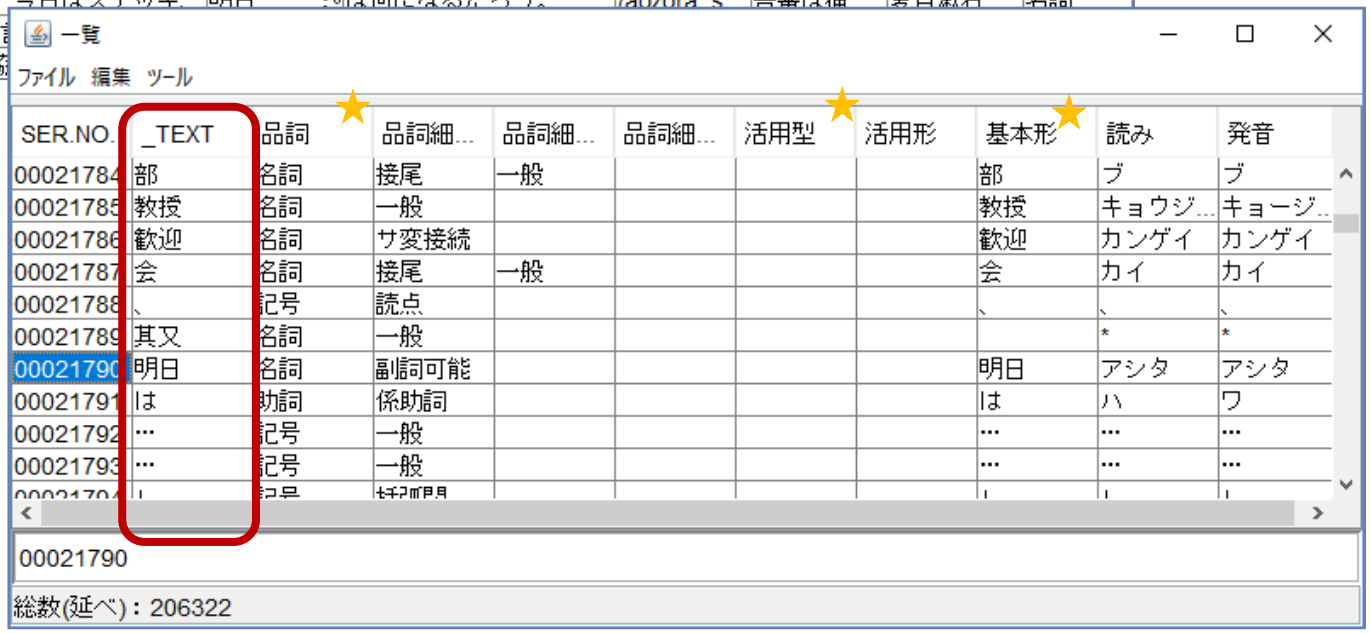


形態素解析結果の閲覧

この機能は，
外部DB「sd」の資料のみ実行可能







テキスト
進行方向

テキストデータのインポートと形態素解析結果のアノテーション

（一般利用者向け）

テキストファイルのインポート
ー青空文庫のテキストデータを例にー

[data]/sample/akutagawa_ryunosuke/kumono_ito.txt

蜘蛛の糸
芥川龍之介

【テキスト中に現れる記号について】

《》:ルビ
(例)蓮池《はすいけ》のふち

青空文庫の独自タグ
(3種類)

｜:ルビの付く文字列の始まりを特定する記号
(例)丁度｜地獄《じごく》の底に

生テキストをインポートする際,
青空文庫のタグは,『ひまわり』用
のタグに変換される(デフォルト)

[＃]:入力者注　主に外字の説明や、傍点の位置の指定
　　(数字は、JIS X 0213の面区点番号、または底本のページと行数)
(例)※[＃「特のへん＋廴＋聿」、第3水準1-87-71]

[＃8字下げ]ー[＃「ー」は中見出し]

　ある日の事でございます。御釈迦様《おしゃかさま》は極楽の蓮池《はすいけ》のふちを、独りでぶらぶら御歩きになっていらっしゃいました。池の中に咲いている蓮《はす》の花は、みんな玉のようにまっ白で、そのまん中にある金色《きんいろ》の蕊《ずい》からは、何とも云えない好《よ》い匂《におい》が、絶間《たえま》なくあたりへ溢《あふ》れて居ります。極楽は丁度朝なのでございましょう。
　やがて御釈迦様はその池のふちに御佇《おたたず》みになって、水の面《おもて》を蔽《おお》っている蓮の葉の間から、ふと下の容子《ようす》を御覧になりました。この極楽の蓮池の下は、丁度｜地獄《じごく》の底に当って居りますから、水晶《すいしよう》のような水を透き徹して、三途《さんず》の河や針の山の景色が、丁度｜覗《のぞ》き眼鏡《めがね》を見るように、はっきりと見えるのでございます。

芥川龍之介: 蜘蛛の糸, 青空文庫(テキストファイル版)

テキストファイルのインポート
ー青空文庫のテキストデータを例にー

蜘蛛の糸
芥川龍之介

【テキスト中に現れる記号について】

《》:ルビ
(例)蓮池《はすいけ》のふち

｜:ルビの付く文字列の始まりを特定する記号
(例)丁度｜地獄《じごく》の底に

［＃］:入力者注　主に外字の説明や、傍点の位置の指定
、または底本のページと行数）
準1-87-71］

［＃8字下げ］ー［＃「ー」は中見出し］

ある日の事でございます。御釈迦様《おしゃかさま》は極楽の蓮池《はすいけ》のふちを、独りでぶらぶら御歩きになっていらっしゃいました。池の中に咲いている蓮《はす》の花は、みんな玉のようにまっ白で、そのまん中にある金色《きんいろ》の蕊《ずい》からは、何とも云えない好《よ》い匂《におい》が、絶間《たえま》なくあたりへ溢《あふ》れて居ります。極楽は丁度朝なのでございましょう。

やがて御釈迦様はその池のふちに御佇《おたたず》みになって、水の面《おもて》を蔽《おお》っている蓮の葉の間から、ふと下の容子《ようす》を御覧になりました。この極楽の蓮池の下は、丁度｜地獄《じごく》の底に当って居りますから、水晶《すいしよう》のような水を透き徹して、三途《さんず》の河や針の山の景色が、丁度｜覗《のぞ》き眼鏡《めがね》を見るように、はっきりと見えるのでございます。

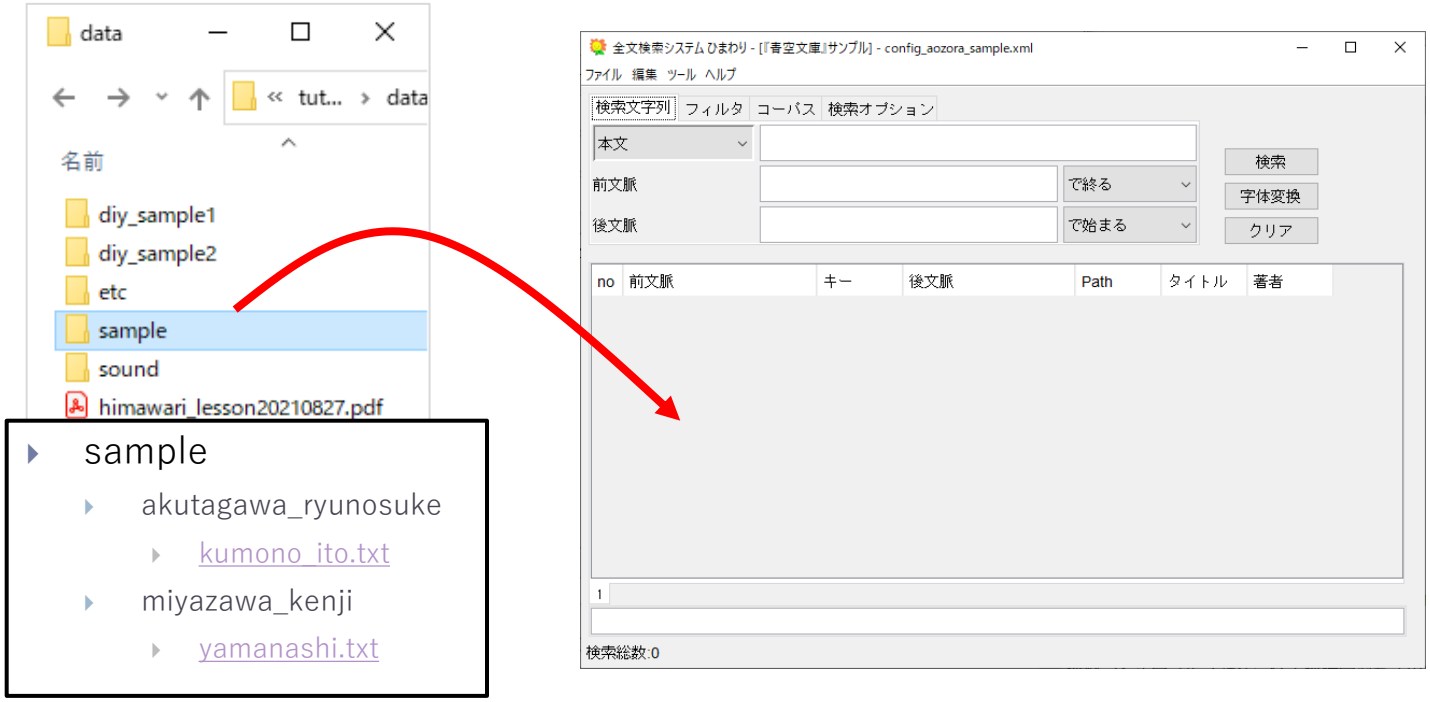
ルビ，注記は，本文とは区別され，全文検索の対象外となる

文字列の照合時は，タグは無視される
（例：「蓮池のふち」にも照合）

芥川龍之介：蜘蛛の糸，青空文庫（テキストファイル版）

インポートの実行

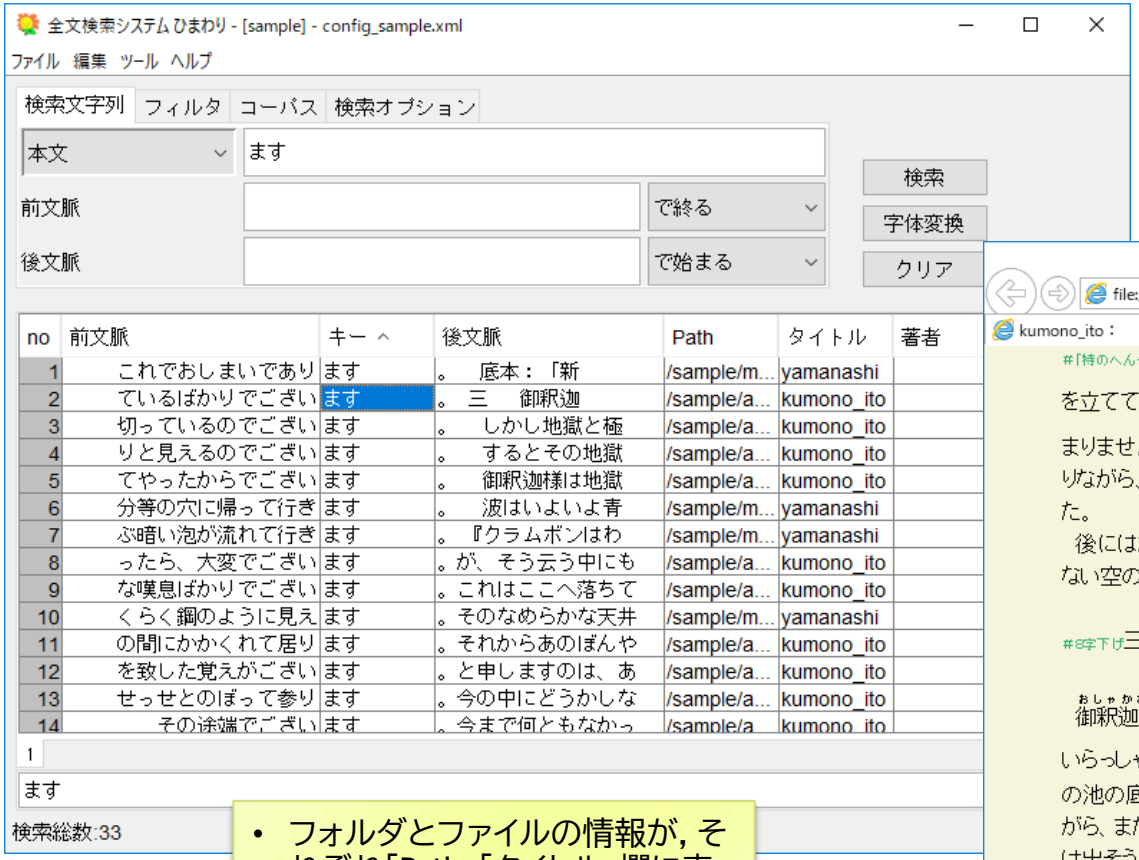
- [data]/sampleフォルダを，起動している『ひまわり』にドラッグ＆ドロップ



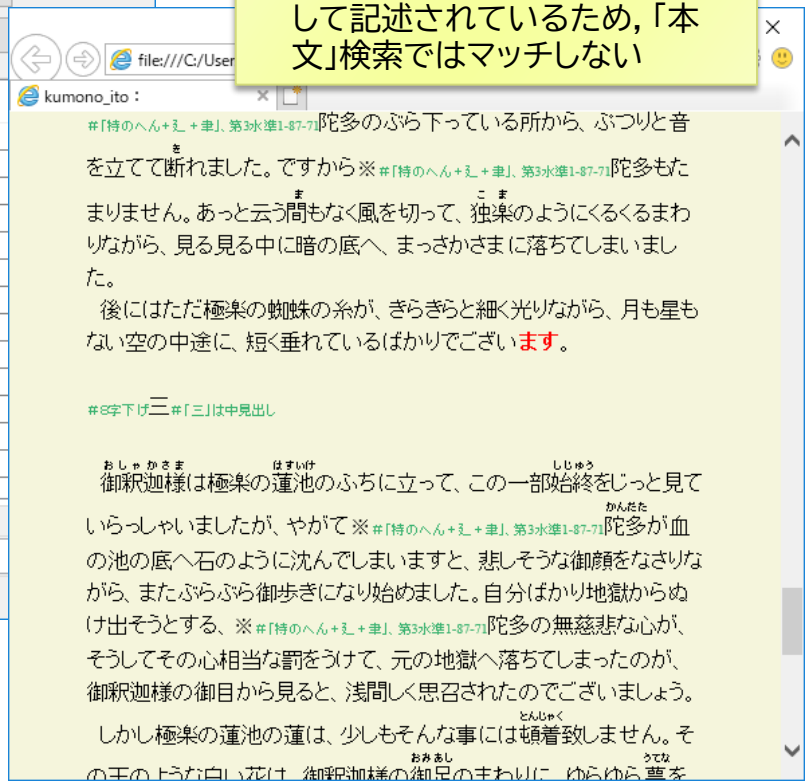
- フォルダの情報をインポート時に利用
  - フォルダ階層 ⇒ Path 欄
  - ファイル名 ⇒ タイトル欄
- ドロップしたフォルダ名がコーパス名になる

- インポート可能なファイル形式(ファイル末尾)
  .txt, .html, .xhtml, .xml
- 文字コードは自動判別
- 詳細オプション(文字列変換，形態素解析など)

検索例

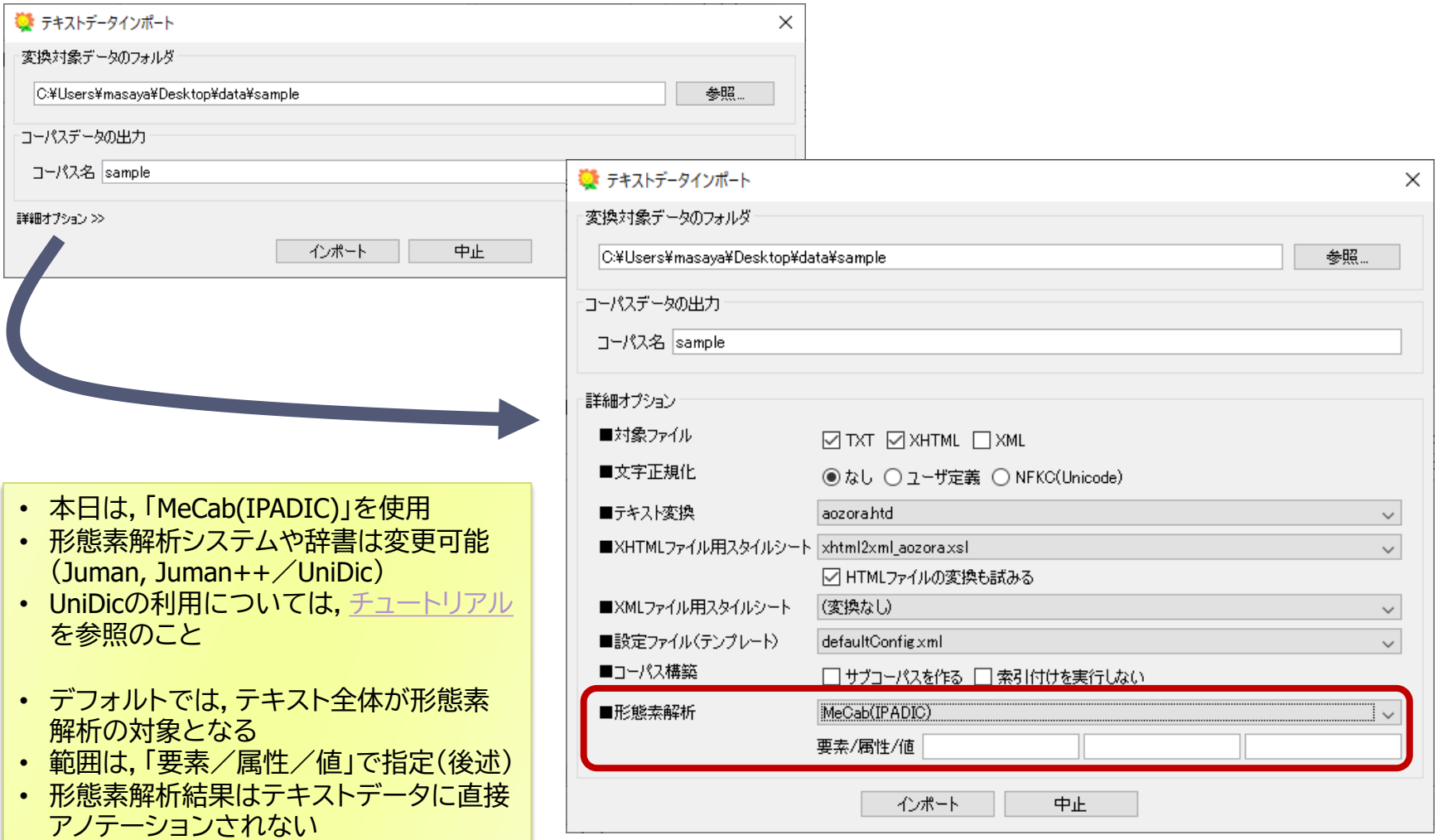


- フォルダとファイルの情報が，それぞれ「Path」「タイトル」欄に表示される
- 「著者」欄は空欄



- ルビ，注記が変換されていることに注目
- ルビ，注記自体はタグの属性として記述されているため，「本文」検索ではマッチしない

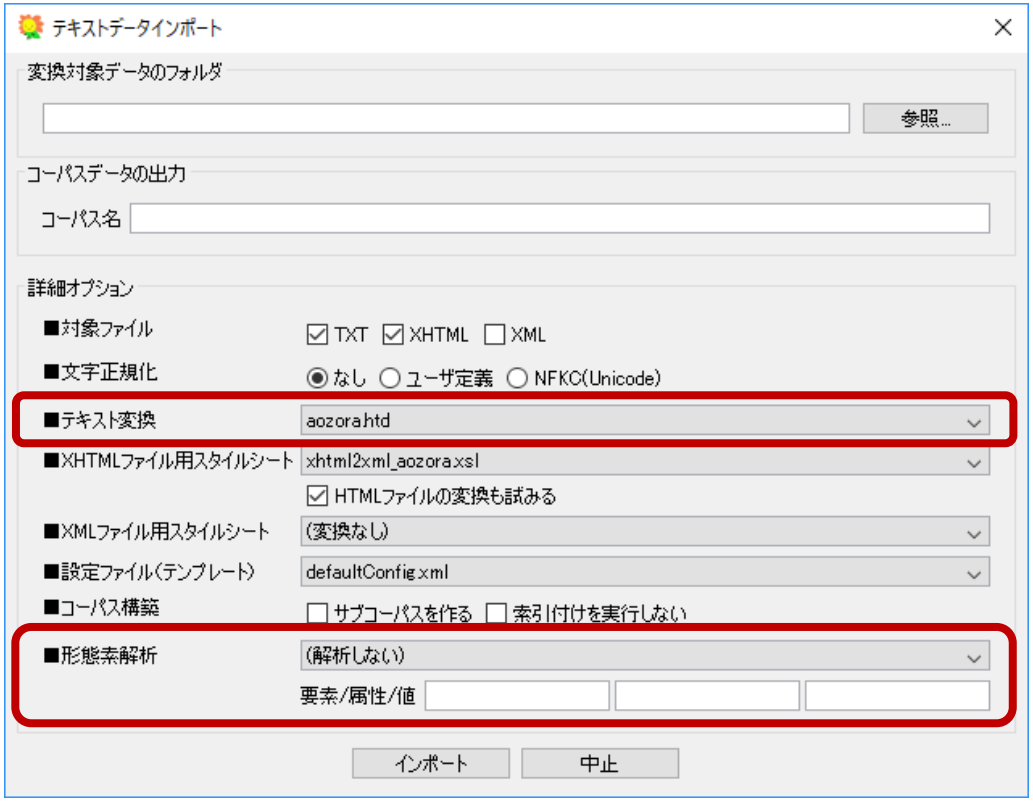
インポート時の形態素解析



- 本日は,「MeCab(IPADIC)」を使用
- 形態素解析システムや辞書は変更可能(Juman, Juman++／UniDic)
- UniDicの利用については, チュートリアルを参照のこと

- デフォルトでは, テキスト全体が形態素解析の対象となる
- 範囲は,「要素／属性／値」で指定(後述)
- 形態素解析結果はテキストデータに直接アノテーションされない

インポート時のオプション



- 文字正規化
  - ユーザ定義： 半角英数字⇒全角
    (.himawari_import_config.xml参照)
  - NFKC: Unicodeで規定される正規化
    - 例： 全角英数字 ⇒ 半角英数字
    - 例： 半角カタカナ ⇒ 全角カタカナ
- テキスト変換
  - [himawari]/resources/htd/aozora.htd
    - 改行位置に,
を挿入
    - 注記, ルビをタグに変換
  - [himawari]/resources/htd/diy.htd
    - 自作コーパス用
    - 汎用タグでテキストにタグ付け可能
- XHTMLファイル用スタイルシート
- XMLファイル用スタイルシート
  - XHTML, XML用の変換規則

『ひまわり』用データの構造と人手アノテーション

（XMLでタグ付け編）

本講習での「人手アノテーション」

- テキストエディタを使い，XMLタグでタグ付けする

```
<t1 arg1=”太郎”>こんにちは。これは，サンプルですか？</t1>
<t1 arg1=”次郎”>そうです。これは，最初のサンプルです。</t1>
```

- 『ひまわり』が直接検索できる形式
- 通常は，人手で作成することはないが仕組みをしるために実施

- テキストエディタを使い，簡易タグでタグ付けする

```
t1(太郎)こんにちは。これは，サンプルですか？/t1
t1(次郎)そうです。これは，最初のサンプルです。/t1
```

- 『ひまわり』用の言語資料を人手で作成しやすくするために「簡易タグ」を利用
- 内部的にXMLへ変換

『ひまわり』用データの構造

- XMLで記述
  - XML(Extensible Markup Language)は, マークアップ言語
  - ここでは, タグを自由に定義できるHTMLのようなものだと考えてください

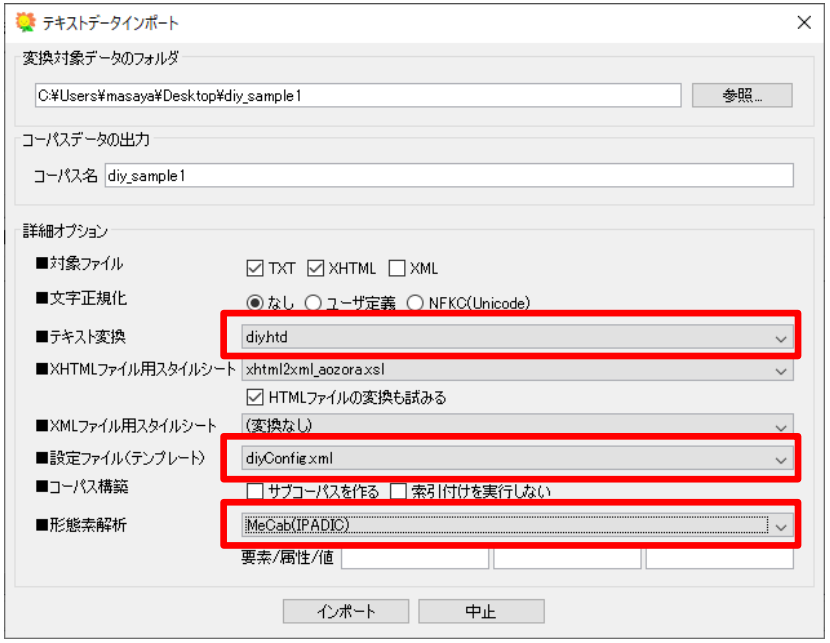
■ 規格化されているため,『ひまわり』以外のツールでも処理可能
■ テキストエディタでも扱える

- 関連するファイルの所在
  - コーパス本体
    - [himawari]/Corpora/コーパス名/corpus.xml
  - 設定ファイル
    - [himawari]/config_コーパス名.xml

■ 使用するタグや検索方法に応じて, 設定ファイルを作成
■ この講習会では, 自作コーパス用の設定を使用

XMLファイルを作ってみる

- [data]/diy_sample1 をインポート
  - diy_sample1フォルダを『ひまわり』にドラッグ＆ドロップしてください
  - 詳細オプションで下記の項目を設定してください
    - テキスト変換: diy.htd
    - 設定ファイル(テンプレート): diyConfig.xml
    - 形態素解析: MeCab(IPADIC)



[data]/diy_sample1/test.txt

太郎　こんにちは。これは，サンプルですか？
次郎　そうです。これは，最初のサンプルです。

■　作成されるファイル
- [himawari]/Corpora/diy_sample1　フォルダ
- [himawari]/config_ diy_sample1.xml

作成されたcorpus.xml

- [himawari]/Corpora/diy_sample1/corpus.xml

```
<?xml version="1.0" encoding="UTF-16"?>
<コーパス 名前="diy_sample1" 備考="script:Himawari ... date:2021-08-22">
<記事 タイトル="test" 著者="" path="/diy_sample1/test.txt" 備考="transDataType:テキスト">
<テキスト>
太郎　こんにちは。これは，サンプルですか？<br />
次郎　そうです。これは，最初のサンプルです。<br />
</テキスト>
</記事>


</コーパス>
```

ファイルを保存するときの注意
- 文字コード：UTF-16
  （little endian, BOM付）
- 改行コード：LF

- 入力ファイル

[data]/diy_sample1/test.txt

```
太郎　こんにちは。これは，サンプルですか？
次郎　そうです。これは，最初のサンプルです。
```

XMLタグの基本

```
<?xml version="1.0" encoding="UTF-16"?>
<コーパス 名前="diy_sample1" 備考="script:Himawari ... date:2021-08-22">
<記事 タイトル="test" 著者="" path="/diy_sample1/test.txt" 備考="transDataType:テキスト">
<テキスト>
太郎　こんにちは。これは，サンプルですか？<br />
次郎　そうです。これは，最初のサンプルです。<br />
</テキスト>
</記事>
</コーパス>
```

ファイル冒頭に，XML宣言を書く

タグには属性を付与できる

- タグ付けの方法
  - 一定範囲に意味づけ
    ⇒ 開始タグと終了タグで囲う
    （例：「コーパス」「記事」「テキスト」タグ ）
  - 特定の位置に意味付け
    ⇒ 空要素タグを使う
    （例：「br」タグ）

- XMLファイル作成時のルール
  - 最上位の要素は一つ
    ⇒　例の場合は，「コーパス」が最上位要素
  - タグの範囲は交差しない
  - メタ文字（半角の＞＜＆など）は代替文字へ変換する

XMLタグの基本(つづき)

- タグの範囲は交差しない

```
<記事>
<テキスト>
        ....
        ....

</テキスト>
</記事>
```

```
<記事>
<テキスト>
        ....
        ....

</記事>
</テキスト>
```

- メタ文字(半角の><&など)は代替文字へ変換する
  - < ⇒ <
  - > ⇒ >
  - & ⇒ &
  - ” ⇒ " (属性部分)
  - ’ ⇒ ' (属性部分)

『ひまわり』のインポート機能では,
文字列検索することを考慮し,
全角文字に変換

タグ付け実習(t1タグ)

- タグ付けの内容
  - 発話部分に対して，t1タグを付与
  - 発話者をt1の属性で表現

- タグ付け前

```
<記事 タイトル="test" 著者="" path="/diy_sample1/test.txt" 備考="transDataType:テキスト">
<テキスト>
太郎　こんにちは。これは，サンプルですか？<br />
次郎　そうです。これは，最初のサンプルです。<br />
</テキスト>
</記事>
```

- タグ付け後

```
<記事 タイトル="test" 著者="" path="/diy_sample1/test.txt" ...>
<テキスト>
<t1 arg1="太郎">こんにちは。これは，サンプルですか？</t1><br />
<t1 arg1="次郎">そうです。これは，最初のサンプルです。</t1><br />
</テキスト>
</記事>
```

タグ付け後の処理

① ファイルの保存

保存時の文字コード, 改行コードに注意！

② XMLファイル形式の検証

開始, 終了タグの対応などをチェック

③ 索引付け

- [ツール] ⇒ [構築] ⇒ インデックス生成 を実行

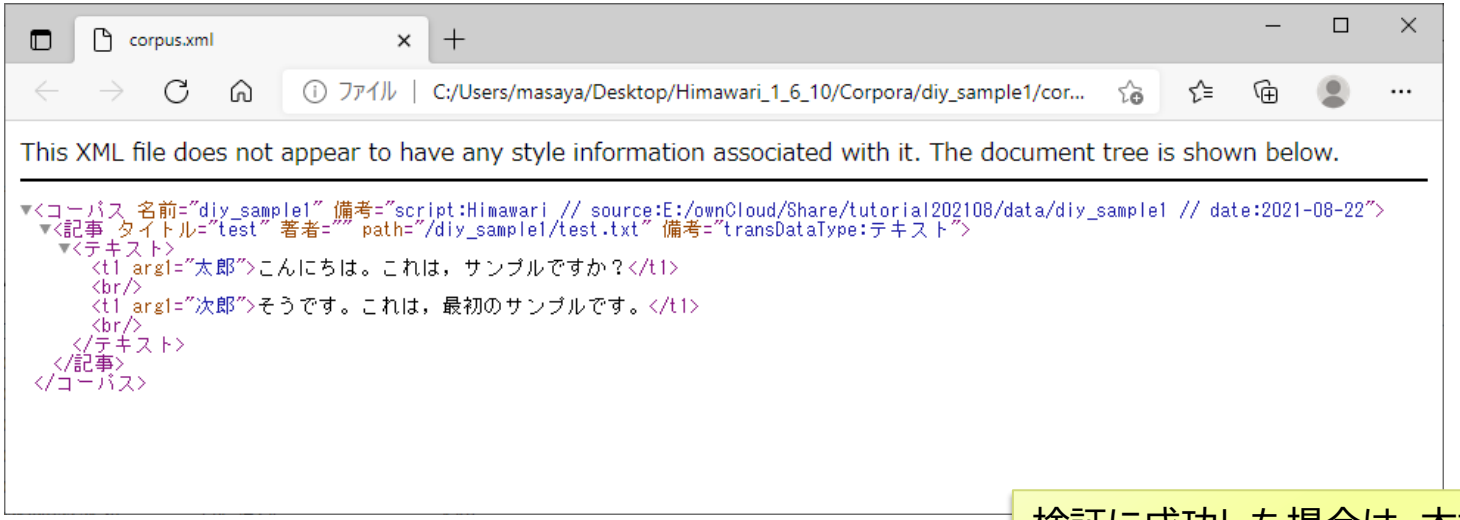
- 全文検索の高速化のために必要な処理
- 設定によっては, 形態素解析も実施

④ （設定ファイルを変更した場合）

- [ファイル] ⇒ [コーパス選択]で設定を再読み込み

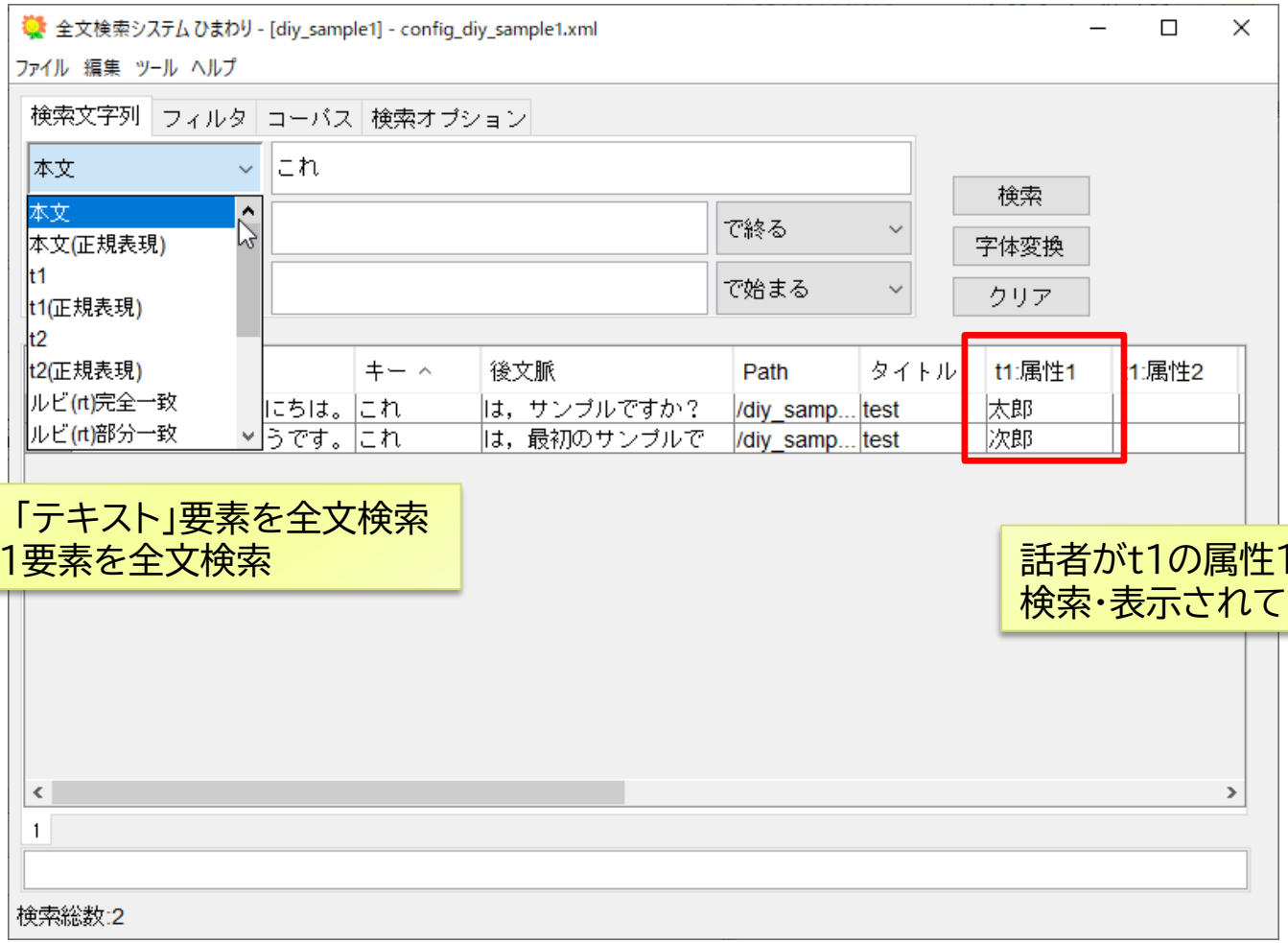
XMLファイル形式の検証（validation）

- 整形式（Well-formed ）のXMLファイルであることを検証
- 小規模なデータであれば，Webブラウザに corpus.xml をドラッグ&ドロップすることで可能
（Chrome, Firefox, Edge（旧版は不可），IEを推奨）



検証に成功した場合は，木構造で表示される

検索例



「本文」：「テキスト」要素を全文検索
「t1」： t1要素を全文検索

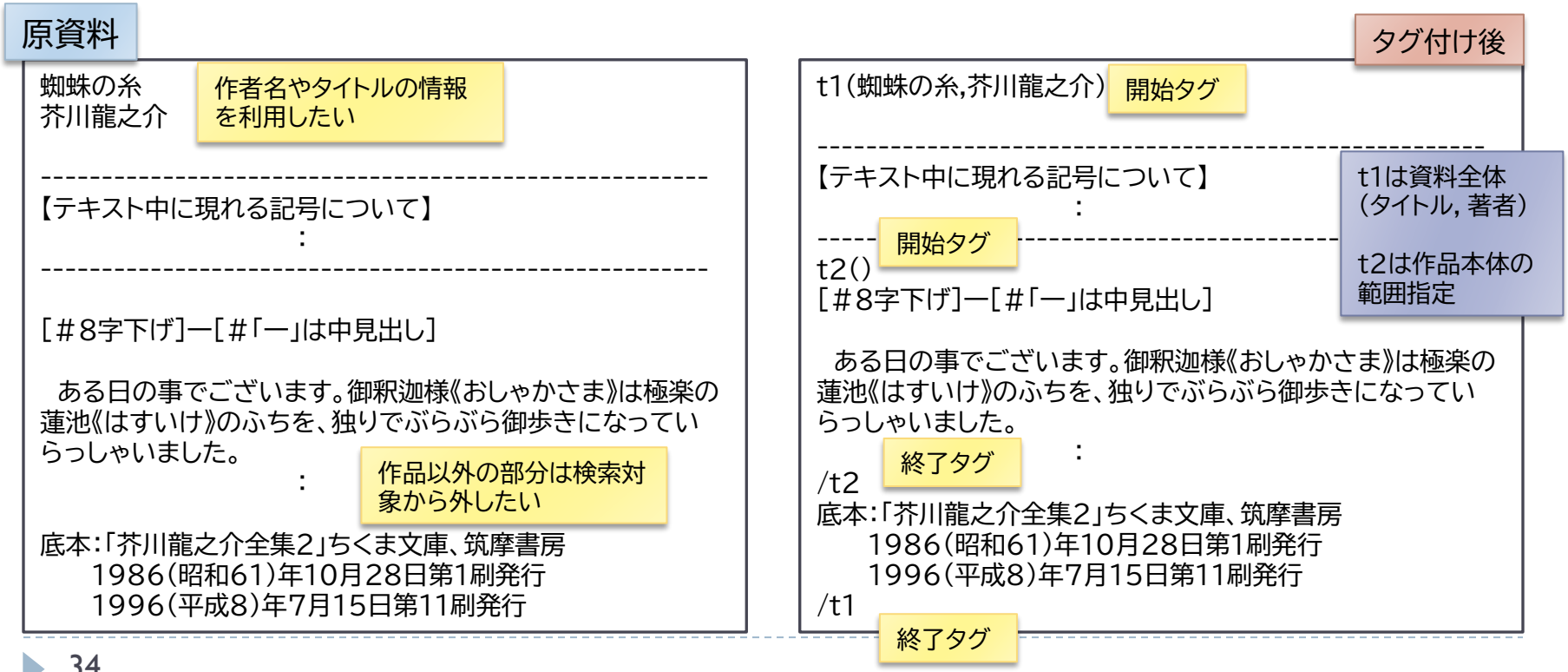
話者がt1の属性1として
検索・表示されている

『ひまわり』用データの構造と人手アノテーション

（簡易タグでタグ付け編）

概要

- 人手でXMLタグを付与するのは大変
- 6種類の簡易タグを使ったアノテーションを行う
  - t1, t2（ブロック要素向け), u1, u2（行内要素向け）
  - e1, e2(空要素)



ブロックレベル要素用タグ t1, t2

- 機能(t1, t2同一)
  - 作品全体, 章や節など, 行以上の範囲をアノテーションするのに使う
  - 開始タグは三つまで属性を持てる

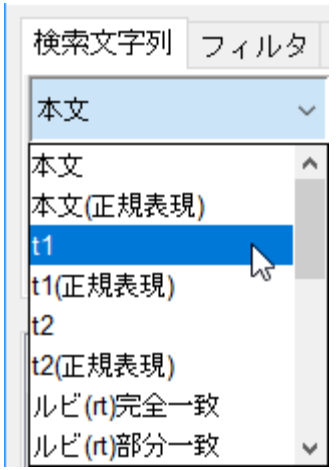
    0個の場合　t1()

    1個の場合　t1(夏目漱石)

    2個の場合　t1(夏目漱石,吾輩は猫である)

    3個の場合　t1(夏目漱石,吾輩は猫である,1905)

❑ 検索対象



t1タグの範囲のみを検索

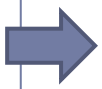
❑ 検索結果

no	前文脈	キー	後文脈	Path	タイトル	t1:属性1	t1:属性2
1		吾輩	は猫である。　名前	/annotatio...	wagahai	夏目漱石	吾輩は猫...

❑ 変換結果のXML

タグは, すべて半角文字

```
t1(夏目漱石,吾輩は猫である)
  吾輩は猫である。名前はまだ無い。
        :
/t1
```

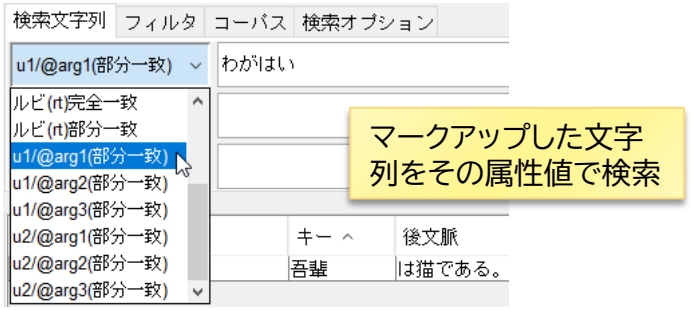


```
<t1 arg1=“夏目漱石” arg2=“吾輩は猫である”>
  吾輩は猫である。名前はまだ無い。
        :
</t1>
```

行内(インライン)要素用タグ u1, u2

- 機能(u1, u2同一)
  - 行内の範囲をアノテーションするのに使用する。
  - 開始タグは三つまで属性を持てる（属性なしは不可）

❑ 検索対象



❑ 青空文庫タグのu1, u2での記述

ルビ

｜二｜疋《ひき》の蟹《かに》の子供らが青じろい水の底で話していました。

→

二u1(ひき)疋/u1のu1(かに)蟹/u1の子供らが青じろい水の底で話していました。

注記

この※[＃「特のへん＋廴＋聿」、第3水準1-87-71]陀多には蜘蛛を助けた事があるのを御思い出しになりました。

→

このu2(特のへん＋廴＋聿,第3水準1-87-71) ※/u2陀多には蜘蛛を助けた事があるのを御思い出しになりました。

空要素タグ e1

- 機能
  - 原資料のページ番号や行位置など，位置を表すのに使う
  - 三つまで属性を持てる（属性なしは不可）
    - e1/（動詞），e1/（動詞,五段），e1/（動詞,五段,未然形）
  - 検索時は，マッチした文字列の先頭文字から見て，文進行方向の最も近いタグの属性値を表示
    - 「吾輩」「吾輩は」「輩」の場合　⇒　「名詞」
    - 「猫である」の場合　⇒　「名詞」

❑ 単語の区切り例

原資料

吾輩は猫である。

タグ付け後

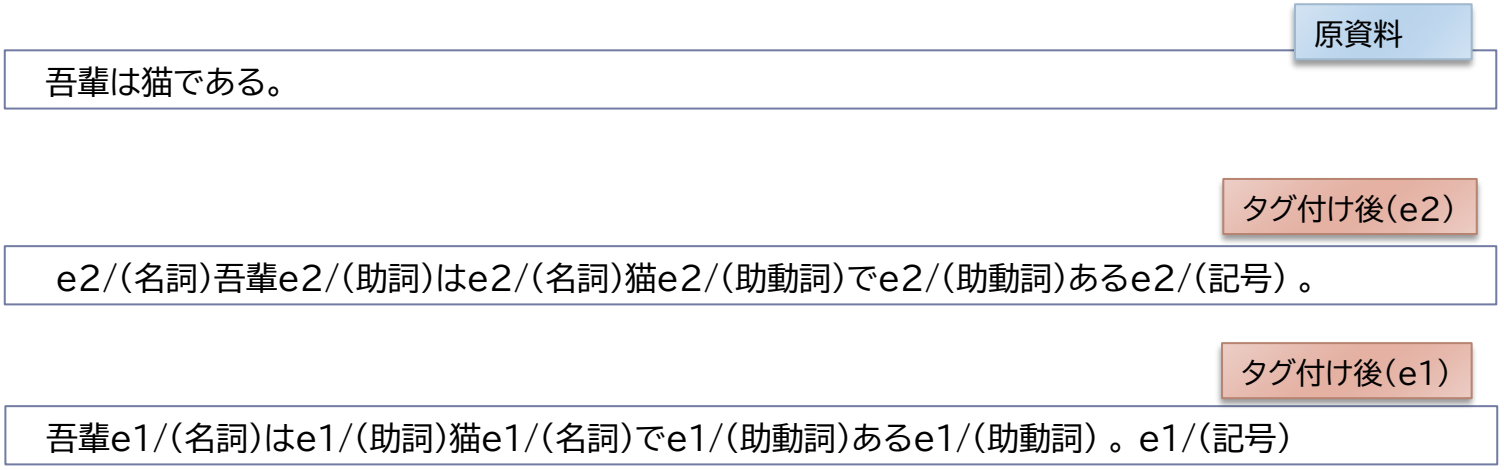
吾輩e1/（名詞）はe1/（助詞）猫e1/（名詞）でe1/（助動詞）あるe1/（助動詞）。e1/（記号）

＊（機能の説明用なので，実用には少し無理がある）

空要素タグ e2

- 機能
  - e1とほぼ同じ機能を持つ
  - ただし, 検索時は, マッチした文字列の先頭文字から見て, 文書先頭方向の最も近いタグの属性値を表示

❑ 単語の区切り例



簡易タグ付け実習

- [data]/diy_sample2 フォルダ

wagahai.txt

【夏目漱石　吾輩は猫である】

吾輩《わがはい》は猫である。【1文目】
名前はまだ無い。【2文目】
どこで生れたかとんと見当《けんとう》がつかぬ。【3文目】

- 全体をt1タグ
- ルビをu1タグ
- 文番号をe1タグ

yamanasi.txt

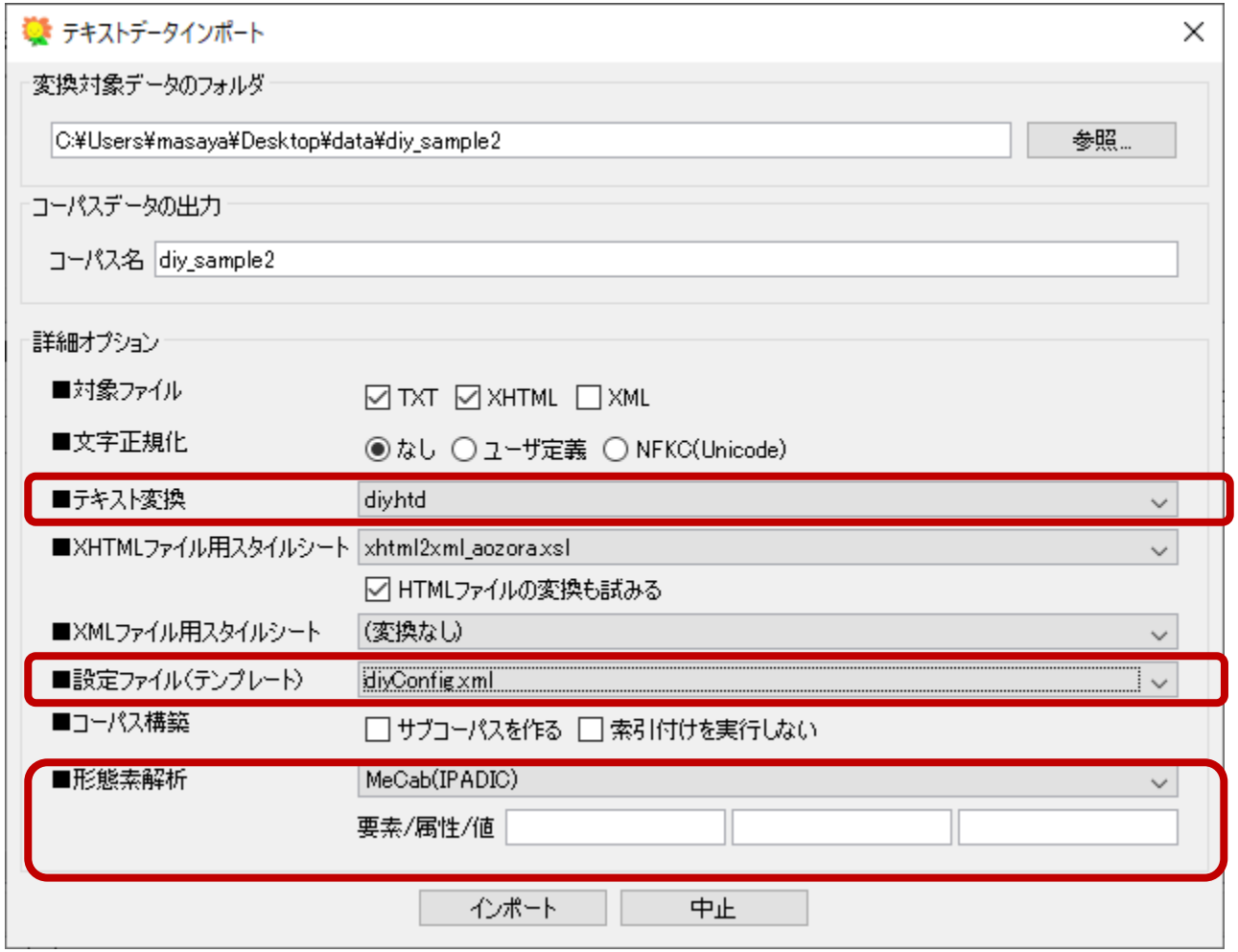
【宮沢賢治　やまなし】

【子供】『お父さん、いまおかしなものが来たよ。』
【お父さん】『どんなもんだ。』
【子供】『青くてね、光るんだよ。はじがこんなに黒く尖ってるの。
それが来たらお魚が上へのぼって行ったよ。』

- 全体をt1タグ
- 1発話をt2タグ

【】内は属性値とし，本文ではないものとする

インポート時のオプション



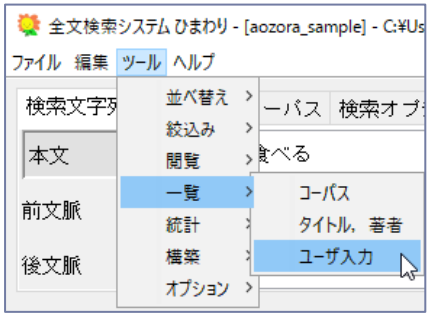
diy.htd

diyConfig.xml

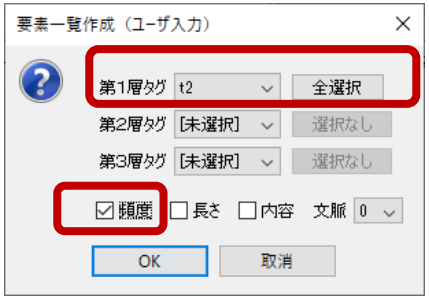
MeCab(IPADIC)

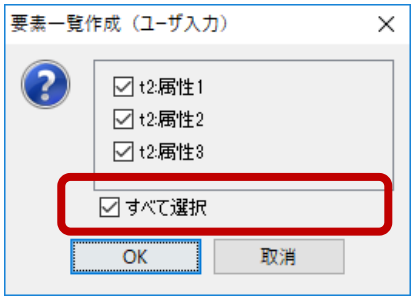
アノテーション結果の集計

- 一覧機能(ユーザ入力)で付与情報を閲覧

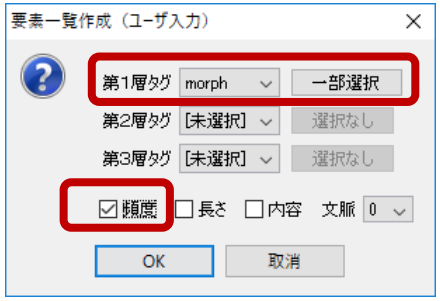


- t2一覧



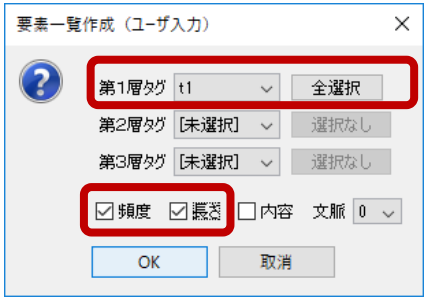


- morph(形態素)一覧



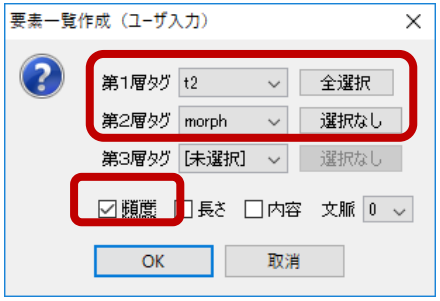
属性
- 品詞
- 品詞分類1～3
- 基本形
- 活用型

- 文字数(t1)



「長さ」:最下位要素の長さ

- t2中の話者別形態素数



タグの包含関係を用いる

『ひまわり』用データの構造と人手アノテーション

（音声データとの対応付け編）

音声データとの関連付け

- 概要
  - 発話データに簡易タグで，発話開始・終了時刻をタグ付け
  - 『ひまわり』の音声プレイヤー機能で再生

- 実習の流れ
  1. 簡易タグ付け
  2. 音声プレイヤーのための設定

簡易タグ付け実習

- [data]/sound/trans_sample フォルダ

announce.txt

```
大変お待たせいたしました。【0.0,2.1】
まもなくスタートいたします。【2.1,4.5】
席に座ってお待ちください。【4.5,7.5】
```

【】内は発話開始時間と発話終了時間(秒)

[data]/sound/wav/announce.wav
「館内アナウンス」, CV: 矢方美紀(Miki Yakata)
https://www.koeyasan.com/

tenki.txt

```
今日の天気は曇り。【0.0,2.0】
すっきりしない一日となるでしょう。【2.0,4.7】
```

[data]/sound/wav/tenki.wav
「天気予報(雨)」, CV: 中元大介(Daisuke Nakamoto)
https://www.koeyasan.com/

- それぞれのファイル全体をt1タグでアノテーション。属性は, 話者名
- 各発話をt2タグでアノテーション。属性は, 発話開始時間と発話終了時間

タグ付け後, 簡易タグ付け実習と同じオプション設定で trans_sample フォルダをインポート

音声プレイヤーのための設定

① [himawari]/config_trans_sample.sd.xml へ次の設定を追加

[data]/sound/sp_setting.txt からコピー可

```
<external_tools>
  <li name="再生" label="再生" field="/^(Path|タイトル)$/"
      path="[[soundplayer]]"
      argument="file:Corpora/trans_sample/wav/((タイトル)).wav ((t2:属性1)) ((t2:属性2))" />
 </external_tools>
```

- 設定ファイルのsetting要素中であれば，どこに追加してもよい
- 「Path」もしくは「タイトル」列の値をクリックすると，プレイヤーを起動
- argument属性の(())は，囲われている列の値で置き換えられる

② 音声データをコピー

- コピー元：[data]/sound/wav
- コピー先：[himawari]/Corpora/trans_sample/wav

おわりに

- 全文検索システム『ひまわり』チュートリアル（作成中心）
  - 『ひまわり』の紹介と基本的な使い方
  - テキストデータのインポート　＋　形態素解析
    （青空文庫作品への自動タグ付け）
  - 『ひまわり』用データの構造と人手アノテーション

- さらに詳しく知るには
  - 『ひまわり』ホームページの各種資料
  - 『ひまわり』用各種パッケージや変換パッケージ
  - スクリプトによるテキスト処理の知識があれば，直接XML形式に変換する方法もあり

参考資料

設定ファイル(config_*.xml)の調整

- 例： 列名を変える

```
 <!-- 結果レコードのフィールド定義 -->
  <field_setting>
    <li align="RIGHT" name="no" type="index" width="30"/>
    <li align="RIGHT" attribute="_preceding_context" element="_sys" name="前文脈" sort_direction="R"
type="preceding_context" width="180"/>
    <li attribute="_key" element="_sys" name="キー" sort_order="1" type="key" width="80"/>
    <li attribute="_following_context" element="_sys" name="後文脈" sort_order="2"
type="following_context" width="160"/>
    <li attribute="path" element="記事" name="Path" type="argument" width="80"/>
    <li attribute="タイトル" element="記事" name="タイトル" type="argument" width="80"/>
    <li attribute="arg1" element="t1" name="t1:属性1" type="argument" width="80"/>
    <li attribute="arg2" element="t1" name="t1:属性2" type="argument" width="80"/>
    <li attribute="arg3" element="t1" name="t1:属性3" type="argument" width="80"/>
    <li attribute="arg1" element="t2" name="t2:属性1" type="argument" width="80"/>
    <li attribute="arg2" element="t2" name="t2:属性2" type="argument" width="80"/>
    <li attribute="arg3" element="t2" name="t2:属性3" type="argument" width="80"/>
```

name属性の値を変更

- 参考資料
  - 『ひまわり』ホームページ⇒「文書」
    ⇒「設定ファイルリファレンスマニュアル」

設定ファイル(config_*.xml)の調整

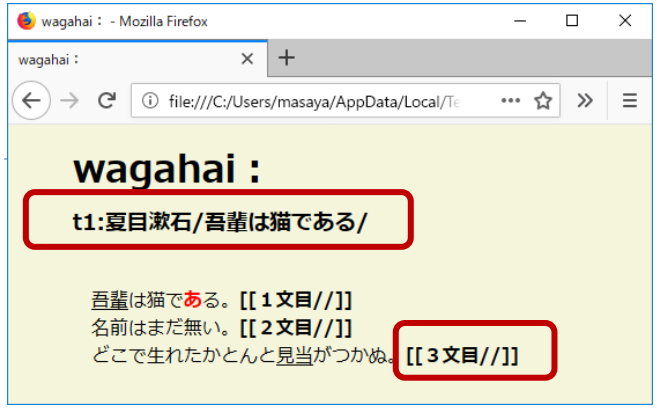
- 全文検索範囲の指定

```
<index_cix>
  <li field_name="キー" label="本文" middle_name="article" name="テキスト" type="normal"/>
  <li field_name="キー" label="本文(正規表現)" middle_name="article" name="テキスト" type="null"/>
  <li field_name="キー" label="t1" middle_name="t11" name="t1" type="normal"/>
  <li field_name="キー" label="t1(正規表現)" middle_name="t12" name="t1" type="null"/>
  <li field_name="キー" label="t2" middle_name="t21" name="t2" type="normal"/>
  <li field_name="キー" label="t2(正規表現)" middle_name="t22" name="t2" type="null"/>
</index_cix>
```

- index_cix/li/@name　　対象とする要素名
- index_cix/li/@type　　全文検索の種類
  - normal：索引付きの通常の全文検索
  - null:　索引なしの正規表現(全文)検索

全文表示用の設定

- 各資料の xsltフォルダ
  - XSLTとCSSの定義ファイル
- annotation_sampleの場合
  - kotobun_written.xsl（XML→HTML変換規則）
  - kotobun_written.css（スタイルシート）



ブラウザとの表示と見比べてみてください

t1用の変換規則

```
<!-- for diy.htd -->
<xsl:template match="t1">
  <h3><xsl:text>t1:</xsl:text><xsl:value-of select="@arg1"/>/<xsl:value-of select="@arg2"/>/<xsl:value-of
select="@arg3"/></h3>
  <xsl:apply-templates/>
</xsl:template>
```

t2用の変換規則

```
<xsl:template match="t2">
  <h3><xsl:text>t2:</xsl:text><xsl:value-of select="@arg1"/>/<xsl:value-of select="@arg2"/>/<xsl:value-of
select="@arg3"/></h3>
  <xsl:apply-templates/>
</xsl:template>
```

e1用の変換規則

```
<xsl:template match="e1">
  <strong><xsl:text>[[</xsl:text><xsl:value-of select="@arg1"/>/<xsl:value-of select="@arg2"/>/<xsl:value-of
select="@arg3"/><xsl:text>]]</xsl:text></strong>
  <xsl:apply-templates/>
</xsl:template>
```

インポート時のテキスト変換(htdファイル)

- 正規表現による文字列置換を利用
  - 正規表現は, Java (クラス Pattern) に準ずる

- 変換規則
  - 『ひまわり』フォルダ/resources/htd に変換規則ファイルを配置
  - 変換規則の形式

    変換前文字列(正規表現) タブ文字 変換後文字列

  - 規則の適用
    - 1入力ファイル全体(改行を含め)を一つの文字列と考える
    - 変換規則を上から順に適用する

変換規則の例
([himawari]/resources/htd フォルダ)

- aozora.htd

```
## 改行位置に，<br />を挿入
¥n          <br />¥n
## 注記
 [(#.+?)]  <注 内容="$1" 付与="" 種別="注記" />
## ルビ(範囲指定あり)
[｜](.+?)《(.+?)》        <r rt="$2">$1</r>
## ルビ(範囲指定なし)
(¥p{InCJKUnifiedIdeographs}+?)《(.+?)》          <r rt="$2">$1</r>
```

※[＃小書き平仮名ん]
⇒ <注 内容="＃小書き平仮名ん" 付与="" 種別="注記" />

一番｜獰悪《どうあく》な
⇒ 一番<r rt="どうあく">獰悪</r>な

蓮池《はすいけ》
⇒ <r rt="はすいけ">蓮池</r>

- diy.htd

```
##5 t1, t2 タグ（ブロックレベル要素，開始タグ）
##例 t1(蜘蛛の糸,芥川龍之介)
##       ⇒ <t1 arg1="蜘蛛の糸" arg2="芥川龍之介">
##
t([12])¥(([^,¥n¥)]*?),([^,¥n¥)]*?),([^¥n¥)]*?)¥)          <t$1 arg1="$2" arg2="$3" arg3="$4">
t([12])¥(([^,¥n¥)]*?),([^¥n¥)]*?)¥) <t$1 arg1="$2" arg2="$3">
t([12])¥(([^¥n¥)]*?)¥)  <t$1 arg1="$2">
t([12])¥(¥) <t$1>

##5 t1, t2 タグ（ブロックレベル要素，終了タグ）
##例 /t1
##       ⇒ </t1>
##
/t([12])    </t$1>
```

t1,t2タグ(開始タグ)の変換
属性の数ごとに定義している

t1 ,t2タグ(終了タグ)の変換

各種設定ファイル＆参考資料

- 『ひまわり』関連
  - 利用者マニュアル
  - 設定ファイルリファレンスマニュアル
  - ビデオチュートリアル
- XSL関連
  - サンプルで覚えるXSLTプログラミング
  - XSLTスタイルシートの基礎の基礎
  - スタイルシート入門（CSS）
- 正規表現関連
  - Java正規表現の使い方
  - Java Pattern クラス