spacyr

---

https://cran.r-project.org/web/packages/spacyr/index.html
https://rdrr.io/cran/spacyr/f/vignettes/using_spacyr.Rmd

Python の spaCy パッケージのラッパー

1. テキストを構文解析し、トークンと文に
2. トークンのレマ化
3. 文法依存関係
4. 連語検索

先に Anaconda と Python をインストールして、
Python で、Spacy をインストールしておく必要がある。

■事前準備

1. Git をインストールする
https://gitforwindows.org/
2. miniconda をインストールする (Miniconda3-latest-Windows-x86_64.exe)
https://www.anaconda.com/download/success

インストール

使用の実際

■サンプルデータ：JPN501.txt from NICER

・本文部分だけにしたテキストファイル： JPN501.txt.data

■使い始め

・英語を指定

■テキスト解析（オプションなしだと、POS 付与）：spacy_parse(txt)

■トークン化（単語ごとにバラバラにする）：spacy_tokenize(txt)

・出力をデータフレームにできる

■「エンティティ」（固有名詞類）の抽出：entity_extract(parsedtxt)

■名詞句抽出：nounphrase_extract(parsedtxt)

・テキスト解析する際に、名詞句オプションを付けておく必要がある。

```
spacy_parse(txt, nounphrase = TRUE)
```

■名詞句をひと塊として POS **タグ付与：**nounphrase_consolidate(parsedtxt)

■エンティティの一覧作成：spacy_extract_entity(txt)

・文中の何単語目か（start_id）も表示

■名詞句のみの一覧作成：spacy_extract_nounphrases(txt)

文法依存関係

■ spacy_parse(**テキスト**， dependency = TRUE, output = "data.frame")

・解析する際に、依存関係のオプションを付ける：dependency = TRUE
・データフレームにしておく：output = "data.frame"

■図にする：textplot

・textplot パッケージを使用
　・UDpipe を前提にしているので、修正必要
・データフレームの見出し pos を upos に名称変更
・一文ずつなので、フィルターで特定の文を選ぶ

パターンマッチング matcher

1. 初期化

```
matcher <- spacy_initialize_matcher()
```

1. パターンの定義

```
pattern <- list( パターンを定義 )
```

1. テキスト解析を施しておく

```
doc <- spacy_parse( テキスト )
```

1. パターンを matcher に追加

```
spacy_add_matcher(matcher, " 文字列 ", pattern)
```

1.matcher の実行

```
result <- spacy_matcher(doc, matcher)
```

1. 結果の出力

```
print(result)
```

終わる前にやっておく：spacy_finalize()