Package 'RmecabKo'

February 13, 2018

Type Package

Title An 'Rcpp' Interface for Eunjeon Project

Version 0.1.6.2

Author Junhewk Kim

Maintainer Junhewk Kim <junhewk.kim@gmail.com>

Description An 'Rcpp' interface for Eunjeon project <http://eunjeon.blogspot.com/>.
The 'mecab-ko' and 'mecab-ko-dic' is based on a C++ library,
and part-of-
speech tagging with them is useful when the spacing of source Korean text is not correct.
This package provides part-of-speech tagging and tokenization function for Korean text.

Imports Rcpp, stringr

LinkingTo Rcpp

License GPL (>= 2)

RoxygenNote 6.0.1

LazyData true

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-02-13 16:11:26 UTC

R topics documented:

install_dic *Install Mecab-Ko-Dic in Linux and Mac OSX.*

Description

`install_dic` installs Mecab-Ko-Dic.

Usage

```
install_dic()
```

Details

This code checks and installs Mecab-Ko-Dic in Linux and Mac OSX. This is essential for using custom-defined user dictionary. Installing Mecab-Ko-Dic needs system previleges, because it uses 'make install' to build from source and install it to system.

Value

None. The function will halt when the current operation system is not Linux or Mac OSX, or Mecab-Ko-Dic is installed already.

See examples in Github.

Examples

```
## Not run:
install_dic()

## End(Not run)
```

install_mecab *Install mecab-ko-msvc and mecab-ko-dic-msvc*

Description

`install_mecab` installs Mecab-Ko-MSVC and Mecab-Ko-Dic-MSVC.

Usage

```
install_mecab(mecabLocation)
```

Arguments

`mecabLocation`	a directory to install Mecab-Ko-MSVC and Mecab-Ko-Dic-MSVC.

Details

This code checks and installs Mecab-Ko-MSVC and Mecab-Ko-Dic-MSVC in user specified directory. Windows only.

Value

None. The function will halt when the current operation system is not Windows, or /mecabLocation/mecab.exe exists.

See examples in Github.

Examples

```
## Not run:
install_mecab("D:/Rlibs/mecab")

## End(Not run)
```

---

nouns *Noun extractor by mecab-ko*

Description

nouns returns nouns extracted from Korean phrases.

Usage

```
nouns(phrase)
```

Arguments

phrase	A character vector or character vectors.

Details

Noun extraction is used for many Korean text analysis algorithms.

Value

List of nouns will be returned. Element name of the list are original phrases.

See examples in Github.

Examples

```
## Not run:
nouns(c("Some Korean Phrases"))

## End(Not run)
```

pos *POS tagging by mecab-ko*

Description

pos returns part-of-speech (POS) tagged morpheme of Korean phrases.

Usage

```
pos(phrase, join = TRUE)
```

Arguments

phrase	Character vector.
join	Boolean.

Details

This is a basic function of part-of-speech tagging by mecab-ko.

Value

List of POS tagged morpheme will be returned in conjoined character vecter form. Element name of the list are original phrases. If join=FALSE, it returns list of morpheme with named with tags.

See examples in Github.

Examples

```
## Not run:
pos(c("Some Korean Phrases"))
pos(c("Some Korean Phrases"), join=FALSE)

## End(Not run)
```

RmecabKo *Rcpp Wrapper for Eunjeon Project*

Description

The mecab-ko and mecab-ko-dic is based on a C++ library, and POS tagging with them is useful when the spacing of source text is not correct. For integrating mecab-ko with R, Rcpp package is used for providing the basic framework.

Details

It is based on the Eunjeon Project. For Mac OSX and Linux, You need to install mecab-ko and mecab-ko-dic before install this package in R. mecab-ko: https://bitbucket.org/eunjeon/mecab-ko mecab-ko-dic: https://bitbucket.org/eunjeon/mecab-ko-dic In Windows, install_mecab(mecabLocati function will install mecab-ko-msvc and mecab-ko-dic-msvc in user specified directory. It is operated by system command and file I/O, the speed of the analysis is slow compared to the Linux-based operating system.

Author(s)

Junhewk Kim

References

- Eunjeon project
- Wonsup Yoon, mecab-ko VC++ builds at https://github.com/Pusnow/mecab-ko-msvc, https://github.com/Pusnow/mecab-ko-dic-msvc

Examples

```
## Not run:
# install.packages("devtools")
devtools::install_github("junhewk/RmecabKo")
# On Windows platform only
install_mecab("D:/Rlibs/mecab")

phrase <- # Some Korean character vectors

# For full POS tagging
pos(phrase)
# For noun extraction only
nouns(phrase)
# For tokenizing of selective morphemes
tokens_words(phrase)
# For n-grams tokenizing
tokens_ngram(phrase)

## End(Not run)
```

---

token_morph *Morpheme tokenizer based on mecab-ko*

Description

These tokernizer functions perform tokenization into full or selected morphemes, nouns.

Usage

```
token_morph(phrase, strip_punct = FALSE, strip_numeric = FALSE)

token_words(phrase, strip_punct = FALSE, strip_numeric = FALSE)

token_nouns(phrase, strip_punct = FALSE, strip_numeric = FALSE)
```

Arguments

phrase	A character vector or a list of character vectors to be tokenized into morphemes. If phrase is a charactor vector, it can be of any length, and each element will be tokenized separately. If phrase is a list of charactor vectors, each element of the list should be a one-item vector.
strip_punct	Bool. If you want to remove punctuations in the phrase, set this as TRUE.
strip_numeric	Bool. If you want to remove numbers in the phrase, set this as TRUE.

Value

A list of character vectors containing the tokens, with one element in the list.

See examples in Github.

Examples

```
## Not run:
txt <- # Some Korean sentence

token_morph(txt)
token_words(txt, strip_punct = FALSE)
token_nouns(txt, strip_numeric = TRUE)

## End(Not run)
```

---

token_ngrams *N-gram tokenizer based on mecab-ko*

Description

This function tokenizes inputs into n-grams. For the developmental purpose, this function offers basic n-gram (or shingle n-gram) only. Other n-gram functionality will be added later. Punctuations and numerics are stripped for this tokenizer, because in Korean n-grams those are usually useless. N-gram function is based on the selective morpheme tokenizer (token_words), but you can select other tokenizer as well.

Usage

```
token_ngrams(phrase, n = 3L, div = c("morph", "words", "nouns"),
  stopwords = character(), ngram_delim = " ")
```

Arguments

phrase	A character vector or a list of character vectors to be tokenized into morphemes. If phrase is a charactor vector, it can be of any length, and each element will be tokenized separately. If phrase is a list of charactor vectors, each element of the list should be a one-item vector.
n	The number of words in the n-gram. This must be an integer greater than or equal to 1.
div	The token generator definition. The options are "morph", "words", and "nouns".
stopwords	Stopwords set to exclude tokens.
ngram_delim	The separator between words in an n-gram.

Value

A list of character vectors containing the tokens, with one element in the list.

See examples in Github.

Examples

```
## Not run:
txt <- # Some Korean sentence

token_ngrams(txt)
token_ngrams(txt, n = 2)

## End(Not run)
```

---

words *Words extractor by mecab-ko*

Description

words returns full morphemes extracted from Korean phrases.

Usage

```
words(phrase)
```

Arguments

phrase	Character vector.

Details

It is based on Mecab-Ko POS classification.

Value

List of full morphemes will be returned.

See examples in Github.

Examples

```
## Not run:
words(c("Some Korean Phrases"))

## End(Not run)
```

Index