Package: RMeCab (via r-universe)

March 5, 2025

Type Package

Title Interface to 'MeCab'

Version 1.14

Maintainer Motohiro Ishida <ishida.motohiro@gmail.com>

Description Parses Japanese texts with 'MeCab'. The original 'MeCab' is licensed under the BSD 3-Clause ``New'' or ``Revised'' License. See the ``LICENSE.note'' file for its license notice.

License GPL (>= 3)

Depends R (>= 4.2)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

SystemRequirements GNU make, MeCab (libmecab-dev (deb), mecab-devel(rpm))

Config/pak/sysreqs make libmecab-dev

Repository https://paithiov909.r-universe.dev

RemoteUrl https://github.com/paithiov909/rmecab-doc

RemoteRef dev

RemoteSha 9099278fad9c81204353a89fe00217d35fd2ff96

Contents

`anyRcfileExists` *anyRcfileExists*

Description

Checks if any mecabrc file exists.

Usage

```
anyRcfileExists()
```

Details

This is a helper function that checks if any mecabrc file exists before initializing tagger.

'MeCab' expects a mecabrc file to be present; if not, it will raise an error (without any message!).

Value

A logical.

`collocate` *collocate*

Description

Finds collocations from the specified text file. Takes a node word and a window span as arguments.

Usage

```
collocate(filename, node, span = 3, dic = "", mecabrc = "", etc = "")
```

Arguments

filename	An input file.
node	Node word.
span	Window span. Defaults to 3.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
out <- collocate(text_file, "\u6570\u5b66")
out

## End(Not run)
```

---

collScores *collScores*

Description

Calculates T-score and MI-score according to the result of collocate().

Usage

```
collScores(kekka, node, span)
```

Arguments

kekka	Result of collocate().
node	Node word.
span	Window span.

Value

A data frame.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
out <- collocate(text_file, "\u6570\u5b66")
collScores(out, "\u6570\u5b66", 3)

## End(Not run)
```

---

docDF *docDF*

Description

Counts tokens (characters, terms, or N-grams) within `target`. `target` can be a file, directory, or a data.frame.

Usage

```
docDF(
  target,
  column = 0,
  type = 0,
  pos = NULL,
  minFreq = 1,
  N = 1,
  Genkei = 0,
  weight = "",
  nDF = 0,
  co = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

`target`	A file, directory, or a data.frame.
`column`	Column number or name which include the text to analyze.
`type`	Kind of tokens. `0` for character, `1` for term. Defaults to `0`.
`pos`	Parts of speech that should be extracted. If `NULL`, all terms are extracted.
`minFreq`	Minimum document frequency for filtering terms. Terms that appear less than `minFreq` within a document are ignored.
`N`	Unit of tokens. If `2`, counts bi-grams.
`Genkei`	If `0`, counts basic form of terms. Defaults to `0`.
`weight`	Method to weight term frequencies.

nDF	If 1, N-grams are divided into columns.
co	If 1, returns co-occurrence matrix.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- docDF(text_dir, column = 0, type = 1, minFreq = 2)
head(out)

## End(Not run)
```

---

docMatrix *docMatrix*

Description

Creates a document-term matrix out of all files in a given directory. Each cell of the matrix shows the actual frequency of each word.

Usage

```
docMatrix(
  mydir,
  pos = "Default",
  minFreq = 1,
  weight = "no",
  kigo = 0,
  co = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

mydir	A directory where text files are stored.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.

weight	Method to weight term frequencies.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
co	If 1, returns co-occurrence matrix.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

An integer matrix is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- docMatrix(text_dir)
head(out)

## End(Not run)
```

---

docMatrix2 *docMatrix2*

Description

Creates a document-term matrix out of all files in a given directory. Each cell of the matrix shows the actual frequency of each word.

Usage

```
docMatrix2(
  directory,
  pos = "Default",
  minFreq = 1,
  weight = "no",
  kigo = 0,
  co = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

directory	A directory where text files are stored or a single file.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.
weight	Method to weight term frequencies.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
co	If 1, returns co-occurrence matrix.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

An integer matrix is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- docMatrix2(text_dir)
head(out)

## End(Not run)
```

---

docMatrixDF *docMatrixDF*

Description

Creates a document-term matrix out of a character vector. Each cell of the matrix shows the actual frequency of each word.

Usage

```
docMatrixDF(
  charVec = c("MeCab", "CaBoCha"),
  pos = "Default",
  minFreq = 1,
  weight = "no",
  co = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

charVec	A character vector.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.
weight	Method to weight term frequencies.
co	If 1, returns co-occurrence matrix.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

An integer matrix is invisibly returned.

---

docNgram *docNgram*

Description

Creates a data.frame of N-gram out of all files in a given directory.

Usage

```
docNgram(
  mydir,
  type = 1,
  N = 2,
  pos = "Default",
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

mydir	A directory where text files are stored.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
N	Unit of tokens. If 2, counts bi-grams.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- docNgram(text_dir, type = 1)
head(out)

## End(Not run)
```

---

docNgram2 *docNgram2*

Description

Creates a data frame of N-grams out of all files in a given directory.

Usage

```
docNgram2(
  directory,
  type = 0,
  pos = "Default",
  minFreq = 1,
  N = 2,
  kigo = 0,
  weight = "no",
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

directory	directory in which text files are stored or a single file.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.
N	Unit of tokens. If 2, counts bi-grams.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
weight	Method to weight term frequencies.

dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- docNgram2(text_dir, type = 1)
head(out)

## End(Not run)
```

---

docNgramDF *docNgramDF*

Description

Creates a data.frame of N-grams out of a character vector.

Usage

```
docNgramDF(
  mojiVec = "MeCab",
  type = 0,
  pos = "Default",
  baseform = 0,
  minFreq = 1,
  N = 1,
  kigo = 0,
  weight = "no",
  co = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

mojiVec	A character vector.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
baseform	Genkei. See docDF(). Defaults to 0.

minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.
N	Unit of tokens. If 2, counts bi-grams.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
weight	Method to weight term frequencies.
co	If 1, returns co-occurrence matrix.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data frame is invisibly returned.

Ngram *Ngram*

Description

Returns a data.frame of N-gram.

Usage

```
Ngram(
  filename,
  type = 0,
  N = 2,
  pos = "Default",
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

filename	An input file.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
N	Unit of tokens. If 2, counts bi-grams.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
out <- Ngram(text_file, type = 1)
head(out)

## End(Not run)
```

NgramDF *NgramDF*

Description

Returns a data frame of N-gram.

Usage

```
NgramDF(
  filename,
  type = 0,
  N = 2,
  pos = "Default",
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

filename	An input file.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
N	Unit of tokens. If 2, counts bi-grams.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
out <- NgramDF(text_file, type = 1)
head(out)

## End(Not run)
```

NgramDF2 *NgramDF2*

Description

Creates a data.frame of N-grams out of all files in a given directory.

Usage

```
NgramDF2(
  directory,
  type = 0,
  pos = "Default",
  minFreq = 1,
  N = 2,
  kigo = 0,
  dic = "",
  mecabrc = "",
  etc = ""
)
```

Arguments

directory	A directory in which text files are stored or a single file.
type	Kind of tokens. 0 for character, 1 for term. Defaults to 0.
pos	Parts of speech that should be extracted. If NULL, all terms are extracted.
minFreq	Minimum document frequency for filtering terms. Terms that appear less than minFreq within a document are ignored.
N	Unit of tokens. If 2, counts bi-grams.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame is invisibly returned.

Examples

```
## Not run:
text_dir <- system.file("samples", package = "RMeCab")
out <- NgramDF2(text_dir, type = 1)
head(out)

## End(Not run)
```

RMeCabC *RMeCabC*

Description

Takes a string as an argument and tokenize it into a length-1 lists of term.

Usage

```
RMeCabC(str, mypref = 0, dic = "", mecabrc = "", etc = "")
```

Arguments

str	A string scalar to be tokenized.
mypref	If 1, returns basic form of terms.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A list.

Examples

```
## Not run:
text <- scan(
  system.file("samples/doc1.txt", package = "RMeCab"),
  what = character()
)
unlist(RMeCabC(text))

## End(Not run)
```

RMeCabDF *RMeCabDF*

Description

Takes a data frame as an argument and tokenize it into a length-1 lists of term.

Usage

```
RMeCabDF(dataf, coln, mypref = 0, dic = "", mecabrc = "", etc = "")
```

Arguments

dataf	A data.frame.
coln	Column number or name which include the text to analyze.
mypref	If 1, returns basic form of terms.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Details

This is a wrapper of RMeCabC(). Any blanks should be replaced with NA for coln.

Value

A list.

RMeCabDoc *RMeCabDoc*

Description

Takes a file as an argument and tokenize it into a list of term.

Usage

```
RMeCabDoc(filename, mypref = 1, kigo = 0, dic = "", mecabrc = "", etc = "")
```

Arguments

filename	An input file.
mypref	If 1, returns basic form of terms.
kigo	If 1, [[TOTAL-TOKENS]] includes number of symbols. Defaults to 0 (does not count symbols).
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A list.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
unlist(RMeCabDoc(text_file))

## End(Not run)
```

RMeCabFreq *RMeCabFreq*

Description

Takes text files as first argument and returns parts of speech and frequencies as a data.frame.

Usage

```
RMeCabFreq(filename, dic = "", mecabrc = "", etc = "")
```

Arguments

filename	an input file.
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A data.frame.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
RMeCabFreq(text_file)

## End(Not run)
```

RMeCabText *RMeCabText*

Description

Takes a file as an argument and tokenize it into a list of terms and parts of speech.

Usage

```
RMeCabText(filename, dic = "", mecabrc = "", etc = "")
```

Arguments

filename	An input file
dic	Path to a user dictionary file such as ishida.dic.
mecabrc	Path to a mecabrc file.
etc	Other options for 'MeCab' tagger.

Value

A list.

Examples

```
## Not run:
text_file <- system.file("samples/doc1.txt", package = "RMeCab")
RMeCabText(text_file)

## End(Not run)
```

Index