Package 'multicastR'

August 30, 2019

Type Package

Title A Companion to the Multi-CAST Collection

Version 1.3.0

URL https://multicast.aspra.uni-bamberg.de/

Description Provides a basic interface for accessing annotation data from the Multi-CAST collection, a database of spoken natural language texts edited by Geoffrey Haig and Stefan Schnell. The collection draws from a diverse set of languages and has been annotated across multiple levels. Annotation data is downloaded on request from the servers of the University of Bamberg. See the Multi-CAST website <https://multicast.aspra.uni-bamberg.de/> for more information and a list of related publications.

License CC BY 4.0

Encoding UTF-8

LazyData true

Depends R (>= 3.0.0), data.table (>= 1.10.0)

Imports stringi (>= 1.1.0), curl (>= 3.3), xml2 (>= 1.1.0), XML (>= 3.98.0), xtable (>= 1.8.0), gsubfn (>= 0.7)

RoxygenNote 6.1.1

Suggests testthat

NeedsCompilation no

Author Nils Norman Schiborr [aut, cre]

Maintainer Nils Norman Schiborr <nils-norman.schiborr@uni-bamberg.de>

Repository CRAN

Date/Publication 2019-08-30 05:50:02 UTC

R topics documented:

mc_clauses 2
mc_index 3

mc_metadata 4
mc_referents 5
mc_table 6
multicast 7
multicastR 9

Index **10**

`mc_clauses` *Count clause units in a multicastR table (WIP)*

Description

Counts the number of clause units (bounded by the <##>, <#>, or <%> annotation symbols) in a multicastR table.

Usage

```
mc_clauses(mcdata, bytext = FALSE)
```

Arguments

`mcdata`	A `data.table` in multicastR format, containing minimally a `corpus` column with the names of the corpora and a `graid` column with GRAID annotation values.
`bytext`	Logical. If `FALSE`, calculate the number of clause units for each corpus. If `TRUE`, count for each text separately.

Value

A `data.table` with the number of valid clause units in each corpus, the total number of clause units, the number of non-analyzed clause units ("NC"), and the percentage the later make up of the total.

See Also

`multicast`, `mc_index`, `mc_referents`, `mc_metadata`

Examples

```
## Not run:
  # count clause units in the most recent version
  # of the Multi-CAST data, by corpus
  n <- mc_clauses(multicast())

  # count by text instead
  m <- mc_clauses(multicast(), bytext = TRUE)

  # number of clauses units in the whole collection
```

```
  sum(n$nClauses)

## End(Not run)
```

`mc_index` *Access the Multi-CAST version index*

Description

`mc_index` downloads an index of versions of the Multi-CAST annotation data from the servers of the Language Archive Cologne (LAC) and outputs it as a `data.table`. The value in the leftmost `version` column may be passed to the `multicast` method for access to earlier versions of the annotations.

Usage

```
mc_index()
```

Value

A `data.table` with five columns:

`[, 1]` `version` Version key. YYMM format. Used for `multicast`'s `vkey` argument.

`[, 2]` `date` Publication date. YYYY-MM-DD format.

`[, 3]` `size` Total file size in kilobytes.

`[, 4]` `texts` Number of texts.

`[, 5]` `corpora` Names of the corpora (languages) included in the version.

See Also

`multicast`.

Examples

```
## Not run:
  # retrieve and print version index
  mc_index()

## End(Not run)
```

mc_metadata *Access the Multi-CAST metadata*

Description

mc_metadata downloads a table with metadata on the texts and speakers in the Multi-CAST collection. The data is downloaded from the servers of University of Bamberg and presented as a data.table.

Usage

```
mc_metadata(vkey)
```

Arguments

vkey	A numeric or character vector of length 1 specifying the requested version of the annotation values. Must be one of the four-digit version keys in the first column of mc_index, or empty. If empty or no value is supplied, the most recent version of the annotations is retrieved automatically.

Value

A data.table containing metadata on the Multi-CAST collection. The table has the following eight columns:

[, 1] corpus The name of the corpus.

[, 2] text The title of the text.

[, 3] type The text type, either TN 'traditional narrative', AN 'autobiographical narrative', or SN 'stimulus-based narrative'.

[, 4] recorded The year (YYYY) the text was recorded.

[, 5] speaker The identifier for the speaker.

[, 6] gender The speaker's gender.

[, 7] age The speaker's age at the time of recording. Approximate values are prefixed with a c.

[, 8] born The speaker's birth year (YYY). Approximate values are prefixed with a c.

See Also

multicast, mc_index, mc_referents, mc_clauses

Examples

```
## Not run:
  # retrieve the most recent version of the Multi-CAST metadata
  mc_metadata()

  # retrieve the lists of referents published in May 2019
```

```
  mc_metadata(1905)   # or: mc_metadata("1905")

  # join the metadata to a table with annotation values
  mc <- multicast()
  merge(mc, mc_metadata(), by = c("corpus", "text"))

## End(Not run)
```

`mc_referents` *Access the Multi-CAST list of referents*

Description

`mc_referents` downloads the lists of referents for all texts in the Multi-CAST collection that have been annotated with the RefIND scheme (Referent Indexing in Natural-language Discourse, Schiborr et al. 2018). The data are downloaded from the servers of University of Bamberg and presented as a `data.table`.

Usage

```
mc_referents(vkey)
```

Arguments

`vkey`	A numeric or character vector of length 1 specifying the requested version of the annotation values. Must be one of the four-digit version keys in the first column of `mc_index`, or empty. If empty or no value is supplied, the most recent version of the annotations is retrieved automatically. Note that the first annotations with RefIND were added with version 1905 (May 2019), and hence no lists of referents exist for earlier versions (i.e. 1505 and 1606).

Value

A `data.table` containing lists of referents for all texts with RefIND annotations in the Multi-CAST collection. The table has the following eight columns:

`[, 1]` `corpus` The name of the corpus.

`[, 2]` `text` The title of the text.

`[, 3]` `refind` The four-digit referent index, unique to each referent in a text.

`[, 4]` `label` The label used for the referent.

`[, 5]` `description` A short description of the referent.

`[, 6]` `class` The semantic class of the referent. Legend: `hum` = human, `anm` = animate, `inm` = inanimate, `bdp` = body part, `mss` = mass, `loc` = location, `tme` = time, `abs` = abstract.

`[, 7]` `relations` Relations of the referent to other referents. Legend: `<` = set member of (partial co-reference), `>` = includes (split antecedence), `M` = part-whole.

`[, 8]` `notes` Annotators' notes on the referent and its properties.

See Also

multicast, mc_index, mc_metadata, mc_clauses

Examples

```
## Not run:
  # retrieve the most recent version of the Multi-CAST lists of referents
  mc_referents()

  # retrieve the lists of referents published in May 2019
  mc_referents(1905)   # or: mc_referents("1905")

  # join the list of referents to a table with annotation values
  mc <- multicast()
  merge(mc, mc_referents(),
        by = c("corpus", "text", "refind"),
        all.x = TRUE)

## End(Not run)
```

---

mc_table *Generate tables with summarized GRAID counts (WIP)*

Description

Constructs simple tables with counts for certain combinations of GRAID form, person/animacy, and function symbols. In its current form, the GRAID categories counted for the tables are predetermined and cannot be changed by the user. The TEX files that can optionally be written by this function are used for the 'Corpus counts' in the Multi-CAST documentation.

Usage

```
mc_table(data, by = "all", format = "wide", write = FALSE,
  writeto = getwd(), output = "tex")
```

Arguments

data	A data.table in multicastR format.
by	Character. "all" places all data in one table, "corpus" generates one table for each corpus, and "text" one table for each text.
format	Unused. Will be used to select between "wide" and "long" table layouts.
write	Logical. If TRUE, writes output to file.
writeto	A directory to which to write output. Defaults to getwd. Ignored if write is FALSE.
output	Unused. Will be used to specify the file format to write as. Currently only TEX output is supported.

Value

A `data.table` with GRAID counts.

Examples

```
## Not run:
  # generate a summary table for the entire collection
  mc <- multicast()
  mc_table(mc)

  # generate a summary table for the English corpus
  mc_table(mc[corpus == "english", ])

## End(Not run)
```

`multicast` *Access Multi-CAST annotation data*

Description

`multicast` downloads the Multi-CAST annotation data from the servers of the University of Bamberg and outputs them as a `data.table`. As the Multi-CAST collection is amenable to extension by additional data sets and annotation schemes, `multicast` takes an optional argument `vkey` to select earlier versions of the annotation data to ensure scientific accountability and reproducibility.

Usage

```
multicast(vkey)
```

Arguments

`vkey`	A numeric or character vector of length 1 specifying the requested version of the annotation values. Must be one of the four-digit version keys in the first column of `mc_index`, or empty. If empty or no value is supplied, the most recent version of the annotations is retrieved automatically. See the examples below for an illustration.

Value

A `data.table` with eleven columns:

`[, 1]` `corpus` The name of the corpus.

`[, 2]` `text` The title of the text. If `legacy.colnames` is `TRUE`, this column is named `file` instead.

`[, 3]` `uid` The utterance identifier. Uniquely identifies an utterance within a text.

`[, 4]` `gword` Grammatical words. The tokenized utterances in the object language. If `legacy.colnames` is `TRUE`, this column is named `word` instead.

`[, 5]` `gloss` Morphological glosses following the Leipzig Glossing Rules.

`[, 6]` `graid` Annotations using the GRAID scheme (Haig & Schnell 2014).

`[, 7]` `gform` The form symbol of a GRAID gloss.

`[, 8]` `ganim` The person-animacy symbol of a GRAID gloss.

`[, 9]` `gfunc` The function symbol of a GRAID gloss.

`[, 10]` `refind` Referent tracking using the RefIND scheme (Schiborr et al. 2018).

`[, 11]` `isnref` Annotations of the information status of newly introduced referents with ISNRef, a simplified version of the RefLex scheme (Riester & Baumann 2017).

Licensing

The Multi-CAST annotation data accessed by the `multicast` method is published under a *Create Commons Attribution 4.0 International* (CC-BY 4.0) licence (https://creativecommons.org/licenses/by-sa/4.0/). Please refer to the collection documentation for information on how to give proper credit to its contributors.

Citing Multi-CAST

Data from the Multi-CAST collection should be cited as:

- Haig, Geoffrey & Schnell, Stefan (eds.). 2015. *Multi-CAST: Multilingual Corpus of Annotated Spoken Texts*. (https://multicast.aspra.uni-bamberg.de/) (Accessed *date*.)

If you need to cite this package specifically, please refer to `citation(multicastR)`.

References

- Haig, Geoffrey & Schnell, Stefan. 2014. *Annotations using GRAID (Grammatical Relations and Animacy in Discourse): Introduction and guidelines for annotators.* Version 7.0. (https://multicast.aspra.uni-bamberg.de/#annotations)
- Riester, Arndt & Baumann, Stefan. 2017. The RefLex scheme – Annotation guidelines. *SinSpeC: Working papers of the SFB 732* 14. (https://dx.doi.org/10.18419/opus-9011))
- Schiborr, Nils N. & Schnell, Stefan & Thiele, Hanna. 2018. *RefIND – Referent Indexing in Natural-language Discourse: Annotation guidelines.* Version 1.1. (https://multicast.aspra.uni-bamberg.de/#annotations)

See Also

`mc_index`, `mc_referents`, `mc_metadata`, `mc_clauses`

Examples

```
## Not run:
  # retrieve and print the most recent version of the
  # Multi-CAST annotations
  multicast()

  # retrieve the version of the annotation data published
  # in May 2019
  multicast(1905)   # or: multicast("1905")
```

```
## End(Not run)
```

`multicastR` *multicastR: A companion to the Multi-CAST collection.*

Description

The `multicastR` package provides a basic interface for accessing annotation data in the Multi-CAST collection (edited by Geoffrey Haig and Stefan Schnell), a database of spoken natural language texts that draws from a diverse set of languages and has been annotated across multiple levels. Annotation data is downloaded on command from the servers of the University of Bamberg via the `multicast` method. Details on the Multi-CAST project and a list of publications can be found online at https://multicast.aspra.uni-bamberg.de/.

Licensing

The Multi-CAST annotation data accessed by the `multicast` method is published under a *Create Commons Attribution 4.0 International* (CC-BY 4.0) licence (https://creativecommons.org/licenses/by-sa/4.0/). Please refer to the collection documentation for information on how to give proper credit to its contributors.

Citing Multi-CAST

Data from the Multi-CAST collection should be cited as:

- Haig, Geoffrey & Schnell, Stefan (eds.). 2018[2015]. *Multi-CAST: Multilinguial Corpus of Annotated Spoken Texts*. (http://multicast.aspra.uni-bamberg.de/) (Accessed *date*.)

If for some reason you need to cite this package specifically, please refer to `citation(multicastR)`.

See Also

`multicast`, `mc_index`, `mc_metadata`, `mc_referents`

Index

data.table, *2–7*

mc_clauses, 2, *4*, *6*, *8*
mc_index, *2*, 3, *4–9*
mc_metadata, *2*, 4, *6*, *8*, *9*
mc_referents, *2*, *4*, 5, *8*, *9*
mc_table, 6
multicast, *2–4*, *6*, 7, *9*
multicastR, 9
multicastR-package *(*multicastR*)*, 9