Package: topomatch (via r-universe)

August 30, 2024

Type Package

Title Match toponyms by similarity

Version 0.4

Author Facundo Muñoz <facundo.munoz@cirad.fr>

Maintainer Facundo Muñoz <facundo.munoz@cirad.fr>

Description Wraps local-global alignment algorithm borrwed from package Biostrings Works better than global alignment and requires less fine-tuning (although is considerably slower too) https://ro-che.info/articles/2016-12-11-local-alignment. Allows to inspect the matching and act accordingly.

Imports Biostrings

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://cirad-astre.r-universe.dev

RemoteUrl https://forgemia.inra.fr/umr-astre/topomatch

RemoteRef HEAD

RemoteSha f61ff809aedea27c545121836bf34e536511d595

Contents

best_matches 2
exact_matches 2
similar_matches 3
topomatch 3
transcribe 4
trim 5
unmatched 6

Index 7

`best_matches` *Best matches*

Description

Best matches

Usage

```
best_matches(x)
```

Arguments

`x`	`topomatch` object

Value

The original list of toponyms with their best-ranked candidates.

Examples

```
best_matches(topomatch(c("a", "b"), c("A", "X")))
```

`exact_matches` *Exact toponym matches*

Description

Exact toponym matches

Usage

```
exact_matches(x)
```

Arguments

`x`	`topomatch` object

Value

named logical vector

Examples

```
exact_matches(topomatch(c("a", "b"), "A"))
```

similar_matches *Similar toponym matches*

Description

Similar toponym matches

Usage

```
similar_matches(x)
```

Arguments

x	topomatch object

Value

named logical vector

Examples

```
similar_matches(topomatch(c("a", "b"), "A"))
```

topomatch *Toponomy matching*

Description

Match a list of toponyms against a list of candidates.

Usage

```
topomatch(x, candidates)
```

Arguments

x	character vector of terms to match, or data.frames with the same number of columns corresponding to different nested levels of administrative units (e.g. region, department, municipality).
candidates	character vector or data.frame of terms to compare against.

Details

For comparison, strings are trimmed off white spaces, converted to upper-case and special characters transliterated to ASCII.

When comparing places defined at multiple hierarchical levels (using multiple columns), the level-wise scores are multiplied.

Value

A matrix with x in rows and candidates in columns with matching scores.

Examples

```
## Match vectors of toponyms
x <- c("abá", "Efé", "IJI", "kkk")
cand <- c("aba", "ef", "iji", "opo", "Uvu")
topomatch(x, cand)

## Match data.frames of toponyms defined at multiple levels
x <- data.frame(
  region = c("A", "A", "B", "B"),
  province = c("abá", "Efé", "IJI", "kkk")
)
cand <- data.frame(
  reg  = c("a", "à", "b", "c", "b"),
  prov = c("aba", "ef", "iji", "opo", "Uvu")
)
topomatch(x, cand)
```

---

transcribe *Transcribe toponyms according to a topomatch table*

Description

Build a function for transcription.

Usage

```
transcribe(obj, fixes = NULL)
```

Arguments

obj	A topomatch object.
fixes	A Named character vector of manual fixes.

Details

If there are any unmatched toponyms, they must be manually matched in fixes for transcription. On the other hand, fixes can provide matches for incorrect associations. See examples.

Value

A function that takes a character vector among the original terms and returns a character vector with the corresponding matches.

Examples

```
(perfect <- topomatch("AFGHANISTAN", "Afghanistan"))
transcribe(perfect)("Afghanistan")

terms <- c("foo", "bar", "foobar", "X")
(mat <- topomatch(terms, c("Foo", "Bar")))
setNames(transcribe(mat, fixes = c(X = "Bar"))(terms), terms)

## You can still transcribe terms that do not need fixes
# transcribe(mat)(terms)  # Error: X is not fixed
transcribe(mat)(c("foo", "bar"))  # Works even if X remains unfixed.
```

---

trim *Trim whitespaces*

Description

Remove the whitespace before or after a string of character

Usage

```
trim(x, internal = TRUE)
```

Arguments

x	character vector, matrix or array.
internal	logical. Whether to replace sequential internal by a single space

Value

A character object of the same type.

Author(s)

Robert J. Hijmans and Jacob van Etten. The code was borrowed from function trim in package raster.

Examples

```
trim(c("    hi folks    !   ", "    hello   world    "))
```

`unmatched` *Unmatched toponyms*

Description

Vector of toponyms with more than one plausible candidate with the same score.

Usage

```
unmatched(x)
```

Arguments

`x`	`topomatch` object

Value

named logical vector

Examples

```
unmatched(topomatch(c("a", "b"), c("A", "X")))
```

Index

best_matches, 2

exact_matches, 2

similar_matches, 3

topomatch, 3
transcribe, 4
trim, *5*, 5

unmatched, 6