Package: apportita (via r-universe)

September 6, 2024

Type Package

Title Utility for Handling 'magnitude' Word Embeddings

Version 0.0.5

Maintainer Akiru Kato <paithiov909@gmail.com>

Description A partial R port from 'magnitude', which is a fast, simple utility library for handling vector embeddings. The main goal of this package is to enable access to user's local magnitude data store.

License MIT + file LICENSE

URL https://github.com/paithiov909/apportita, https://paithiov909.github.io/apportita/

BugReports https://github.com/paithiov909/apportita/issues

Imports dbplyr, dplyr, methods, proxyC, purrr, rlang, RSQLite, stats, tibble, transport, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.1

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

RemoteUrl https://github.com/paithiov909/apportita

RemoteRef HEAD

RemoteSha 7e1b28199c8630583f4ac38a89820105501ae6a7

Contents

`calc_dist` *Calculate distances from keys to keys*

Description

Calculate distances from keys to keys

Usage

```
calc_dist(
  conn,
  keys,
  q,
  normalized = TRUE,
 method = c("euclidean", "chisquared", "kullback", "manhattan", "maximum", "canberra",
    "minkowski", "hamming"),
  ...
)
```

Arguments

`conn`	a Magnitude connection.
`keys`	character vector.
`q`	character vector.
`normalized`	logical; whether or not vector embeddings should be normalized?
`method`	string; method to compute distance.
`...`	other arguments are passed to `proxyC::dist`.

Value

a sparse Matrix of 'Matrix' package.

calc_simil — *Calculate similarities from keys to keys*

Description

Calculate similarities from keys to keys

Usage

```
calc_simil(
  conn,
  keys,
  q,
  normalized = TRUE,
 method = c("cosine", "correlation", "jaccard", "ejaccard", "dice", "edice", "hamann",
    "simple matching", "faith"),
  ...
)
```

Arguments

`conn`	a Magnitude connection.
`keys`	character vector.
`q`	character vector.
`normalized`	logical; whether or not vector embeddings should be normalized?
`method`	string; method to compute similarity.
`...`	other arguments are passed to `proxyC::simil`.

Value

a sparse Matrix of 'Matrix' package.

calc_wrd — *Calculate Word Rotator's Distance from keys to keys*

Description

Calculate Word Rotator's Distance from keys to keys

Usage

```
calc_wrd(conn, keys, q, normalized = TRUE, ...)
```

Arguments

`conn`	a Magnitude connection.
`keys`	character vector.
`q`	character vector.
`normalized`	logical; whether or not vector embeddings should be normalized?
`...`	other arguments are passed to `transport::wasserstein` internally.

Value

numeric scalar.

`close,Magnitude-method` *Close a Magnitude connection*

Description

Close a Magnitude connection

Usage

```
## S4 method for signature 'Magnitude'
close(con)
```

Arguments

`con`	a Magnitude connection.

Value

the value from `RSQLite::dbDisconnect` is returned invisibly.

`dim,Magnitude-method` *Dimensions of a Magnitude table*

Description

Dimensions of a Magnitude table

Usage

```
## S4 method for signature 'Magnitude'
dim(x)
```

Arguments

x	a Magnitude connection.

Value

a numeric vector.

---

doesnt_match *Order keys by their distances to a key*

Description

Order keys by their distances to a key

Usage

```
doesnt_match(
  conn,
  key,
  q,
  n = 1L,
  normalized = TRUE,
 method = c("euclidean", "chisquared", "kullback", "manhattan", "maximum", "canberra",
    "minkowski", "hamming")
)
```

Arguments

conn	a Magnitude connection.
key	string.
q	character vector. elements exact same with key will be dropped from result.
n	integer.
normalized	logical; whether or not vector embeddings should be normalized?
method	string; method to compute distance.

Value

a tibble.

has_exact — *Check if keys exist in a Magnitude table?*

Description

Check if keys exist in a Magnitude table?

Usage

```
has_exact(conn, keys)
```

Arguments

conn	a Magnitude connection.
keys	a character vector.

Value

a tibble.

magnitude — *Create a Magnitude connection*

Description

Create a Magnitude connection

Usage

```
magnitude(path, ...)
```

Arguments

path	string; a path to a magnitude file.
...	other arguments are passed to `RSQLite::dbConnect`.

Value

a Magnitude connection object inheriting SQLiteConnection class from 'RSQLite' package.

most_similar — *Order keys by their similarity to a key*

Description

Order keys by their similarity to a key

Usage

```
most_similar(
  conn,
  key,
  q,
  n = 1L,
  normalized = TRUE,
 method = c("cosine", "correlation", "jaccard", "ejaccard", "dice", "edice", "hamann",
    "simple matching", "faith")
)
```

Arguments

conn	a Magnitude connection.
key	string.
q	character vector. elements exact same with key will be dropped from result.
n	integer.
normalized	logical; whether or not vector embeddings should be normalized?
method	string; method to compute similarity.

Value

a tibble.

query — *Get vector embeddings of keys*

Description

Get vector embeddings of keys. If out of vocabulary, their embeddings would be generated at random.

Usage

```
query(
  conn,
  q,
  normalized = TRUE,
  ngram_beg = NULL,
  ngram_end = NULL,
  topn = 5L
)
```

Arguments

conn	a Magnitude connection.
q	a character vector.
normalized	logical; whether or not vector embeddings should be normalized?
ngram_beg	integer. If supplied, the function gets out-of-vocabulary vectors by using character ngrams of which length are ‘ngram_end - ngram_beg‘.
ngram_end	integer.
topn	integer used for making out-of-vocabulary vectors.

Value

a tibble.

---

slice_frac *Slice samples by fraction from a Magnitude table*

Description

Slice samples by fraction from a Magnitude table

Usage

```
slice_frac(conn, frac = 0.001, normalized = TRUE)
```

Arguments

conn	a Magnitude connection.
frac	numeric.
normalized	logical; whether or not vector embeddings should be normalized?

Value

a tibble.

slice_index — *Slice samples by index from a Magnitude table*

Description

Slice samples by index from a Magnitude table

Usage

```
slice_index(conn, index, normalized = TRUE)
```

Arguments

conn	a Magnitude connection.
index	integer vector.
normalized	logical; whether or not vector embeddings should be normalized?

Value

a tibble.

slice_n — *Slice samples from a Magnitude table*

Description

Slice samples from a Magnitude table

Usage

```
slice_n(conn, n, offset = 0, normalized = TRUE)
```

Arguments

conn	a Magnitude connection.
n	integer.
offset	integer.
normalized	logical; whether or not vector embeddings should be normalized?

Value

a tibble.

`wrd` *Calculate Word Rotator's Distance*

Description

Calculate Word Rotator's Distance between two distributions.

Usage

```
wrd(x, y, ...)
```

Arguments

`x`	a dense or sparse matrix.
`y`	a dense or sparse matrix.
`...`	other arguments are passed to `transport::wasserstein` interenally.

Details

Word Rotator's Distance is a measure of textual similarity improved of Word Mover's Distance.

Value

numeric scalar.

See Also

http://dx.doi.org/10.18653/v1/2020.emnlp-main.236

Index