Package ‘oglcnac’

February 19, 2025

Title Processing and Analysis of 'O-GlcNAcAtlas' Data

Version 0.1.5

Description Provides tools for processing and analyzing data from the 'O-GlcNAcAtlas' database <https://oglcnac.org/>,
as described in Ma (2021) <doi:10.1093/glycob/cwab003>. It integrates 'UniProt' <https://www.uniprot.org/> API calls
to retrieve additional information. It is specifically designed for research workflows
involving 'O-GlcNAcAtlas' data, providing a flexible and user-friendly interface for customizing
and downloading processed results. Interactive elements allow users to easily adjust parameters
and handle various biological datasets.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

Imports cli, httr, jsonlite, shiny, DT, bslib, readxl, glue

Suggests testthat (>= 3.0.0), tibble, curl

Config/testthat/edition 3

NeedsCompilation no

Author Yaoxiang Li [aut, cre] (<https://orcid.org/0000-0001-9200-1016>)

Maintainer Yaoxiang Li <liyaoxiang@outlook.com>

Repository CRAN

Date/Publication 2025-02-19 09:30:02 UTC

Contents

compare_tibbles_uniprot 2
launch_app 3
parse_uniprot_data 3
process_tibble_uniprot 4
retrieve_uniprot_data 5

Index **7**

`compare_tibbles_uniprot`

Compare Input and Updated Tibbles

Description

This function compares the original input tibble and the updated tibble, identifying and reporting any changes in the specified columns (‘entry_name‘, ‘protein_name‘, ‘gene_name‘).

Usage

```
compare_tibbles_uniprot(
  original_tibble,
  updated_tibble,
  entry_name_col = "entry_name",
  protein_name_col = "protein_name",
  gene_name_col = "gene_name"
)
```

Arguments

`original_tibble`	The original tibble before processing.
`updated_tibble`	The tibble returned after processing.
`entry_name_col`	The column name for entry names (default: "entry_name").
`protein_name_col`	The column name for protein names (default: "protein_name").
`gene_name_col`	The column name for gene names (default: "gene_name").

Value

None. Prints the differences between the tibbles.

Examples

```
# Example usage:

# Original input tibble
input_data <- tibble::tibble(
  id = c(1, 2),
  species = c("mouse", "rat"),
  sample_type = c("brain", "liver"),
  accession = c("O88737", "Q9R064"),
  accession_source = c("UniProt", "UniProt")
)

# Process the tibble (this will add the entry_name, protein_name, and gene_name)
```

```
processed_data <- process_tibble_uniprot(input_data)

# Compare the original and processed tibbles
compare_tibbles_uniprot(input_data, processed_data)
```

`launch_app` *Launch oglcnac Shiny App*

Description

This function launches a Shiny App for uploading, processing, and downloading UniProt data in CSV, TSV, or Excel format. Users can upload data, preview it, and select specific columns for processing. The processed data can be viewed and downloaded.

Usage

```
launch_app()
```

Value

None

Examples

```
if (interactive()) {
  oglcnac::launch_app()
}
```

`parse_uniprot_data` *Parse UniProt Data*

Description

This function parses the data retrieved from the UniProt API to extract the entry name, protein name, and gene name.

Usage

```
parse_uniprot_data(uniprot_data)
```

Arguments

`uniprot_data`	A list returned by the UniProt API query.

Value

A list containing ‘entry_name‘, ‘protein_name‘, and ‘gene_name‘.

Examples

```
# Example usage:

# Retrieve UniProt data
test_result <- retrieve_uniprot_data("O88737")

# Parse the UniProt data
parsed_result <- parse_uniprot_data(test_result)

# Print the parsed result
print(parsed_result)
```

process_tibble_uniprot

Process a Tibble of UniProt Data

Description

This function processes a tibble containing accession and accession_source columns. It retrieves data from the UniProt API for rows with accession_source == "UniProt" and overwrites or creates the entry_name, protein_name, and gene_name columns only if the parsed values are not NULL or NA.

Usage

```
process_tibble_uniprot(
  data,
  accession_col = "accession",
  accession_source_col = "accession_source",
  entry_name_col = "entry_name",
  protein_name_col = "protein_name",
  gene_name_col = "gene_name"
)
```

Arguments

`data` A tibble containing at least accession and accession_source columns.

`accession_col` The column name for accession numbers (default: "accession").

`accession_source_col`
The column name for accession sources (default: "accession_source").

`entry_name_col` The column name for entry names (default: "entry_name").

`protein_name_col`
The column name for protein names (default: "protein_name").

`gene_name_col` The column name for gene names (default: "gene_name").

Value

A tibble with UniProt data processed.

Examples

```
# Example usage:

# Load necessary library
library(tibble)

# Reduced example data as an R tibble
test_data <- tibble::tibble(
  id = c(1, 78, 83, 87),
  species = c(”mouse”, ”mouse”, ”rat”, ”mouse”),
  sample_type = c(”brain”, ”brain”, ”brain”, ”brain”),
  accession = c(”O88737”, ”O35927”, ”Q9R064”, ”P51611”),
  accession_source = c(”OtherDB”, ”UniProt”, ”UniProt”, ”UniProt”),
  entry_name = c(”BSN_MOUSE”, NA, ”GORS2_RAT”, NA),
  protein_name = c(”Protein bassoon”, NA, ”Golgi reassembly-stacking protein2”, NA),
  gene_name = c(”Bsn”, NA, ”Gorasp2”, NA)
)

# Process the tibble
result_data <- process_tibble_uniprot(test_data)

# Compare the original and processed tibbles
compare_tibbles_uniprot(test_data, result_data)
```

`retrieve_uniprot_data` *Retrieve Data from UniProt API*

Description

This function sends a GET request to the UniProt REST API and retrieves data based on the provided UniProt accession number.

Usage

```
retrieve_uniprot_data(accession)
```

Arguments

`accession`	A character string representing the UniProt accession number.

Value

A list containing the retrieved data in JSON format, or NULL if the request fails.

Examples

```
# Example usage

result <- retrieve_uniprot_data("O88737")
print(result)
```

Index

compare_tibbles_uniprot, 2

launch_app, 3

parse_uniprot_data, 3
process_tibble_uniprot, 4

retrieve_uniprot_data, 5