Package 'dbparser'

May 8, 2026

Title Drugs Databases Parser

Version 2.2.1

Description This tool is for parsing public drug databases such as 'DrugBank' XML database <https://go.drugbank.com/>.
The parsed data are then returned in a proper 'R' object called 'dvobject'.

License MIT + file LICENSE

Encoding UTF-8

Imports data.table, dplyr, progress, purrr, tibble, tools, utils, XML

RoxygenNote 7.3.3

Suggests canvasXpress, knitr, rmarkdown, testthat, tidyr

VignetteBuilder knitr

URL https://docs.ropensci.org/dbparser/,
https://github.com/ropensci/dbparser

BugReports https://github.com/ropensci/dbparser/issues

Depends R (>= 3.5)

NeedsCompilation no

Author Mohammed Ali [aut, cre],
Ali Ezzat [aut],
Hao Zhu [rev],
Emma Mendelsohn [rev]

Maintainer Mohammed Ali <moh_fcis@yahoo.com>

Repository CRAN

Date/Publication 2026-01-08 22:50:02 UTC

Contents

`add_database_info` *add_database_info Assign passed databases db metadata to passed dvobject*

Description

add_database_info Assign passed databases db metadata to passed dvobject

Usage

```
add_database_info(
  dvobject,
  db_type = "DrugBank",
  db_version = NULL,
  db_exported_date = NULL
)
```

Arguments

`dvobject`	dvobject to assign metadata to it
`db_type`	database type (defualt="DrugBank")
`db_version`	database version as string
`db_exported_date`	database official export date

Value

dvobject

See Also

Other utility: `show_dvobject_metadata()`, `subset_drugbank_dvobject()`, `subset_onsides_dvobject()`

cett_nodes_options *returns carriers, enzymes,targets and transporters node valid options.*

Description

returns carriers, enzymes,targets and transporters node valid options.

Usage

```
cett_nodes_options()
```

Value

list of CETT valid options

See Also

Other parsers: `drug_node_options()`, `parseDrugBank()`, `parseOnSIDES()`, `parseTWOSIDES()`, `references_node_options()`

drug_node_options *returns drug node valid options.*

Description

returns drug node valid options.

Usage

```
drug_node_options()
```

Value

list of drug valid options

See Also

Other parsers: `cett_nodes_options()`, `parseDrugBank()`, `parseOnSIDES()`, `parseTWOSIDES()`, `references_node_options()`

merge_drugbank_onsides *Merge DrugBank and OnSIDES Database Objects*

Description

Creates an integrated dvobject object by linking DrugBank dvobject with OnSIDES dvobject using RxNorm CUIs as the bridge.

Usage

```
merge_drugbank_onsides(db_object, onsides_db)
```

Arguments

db_object	A dvobject from 'parseDrugBank()' OR an existing merged dvobject (containing '$drugbank').
onsides_db	A dvobject produced by 'dbparser::parseOnSIDES()'.

Details

This function performs the following key steps: 1. Creates a mapping table between DrugBank IDs and RxNorm CUIs from the DrugBank object. 2. Enriches the relevant OnSIDES tables ('vocab_rxnorm_ingredient' and optionally 'high_confidence') by adding a 'drugbank_id' column. 3. Assembles a new list object containing all original tables plus the enriched ones and the ID mapping table itself.

The resulting object allows for powerful queries that span both mechanistic data from DrugBank and clinical side-effect data from OnSIDES. Supports piping and chaining with other merge functions.

Value

A new dvobject containing the integrated data.

See Also

Other mergers: merge_drugbank_twosides()

Examples

```
## Not run:
# First, parse the individual databases
drugbank <- parseDrugBank("path/to/drugbank.xml")
onsides <- parseOnSIDES("path/to/onsides_csvs/")

# Now, merge them into a single, powerful object
merged_db <- merge_drugbank_onsides(drugbank, onsides)

# --- Example Analysis: Find the protein targets of all drugs known to ---
```

```
# --- cause the side effect "Hepatitis" with high confidence.        ---

# 1. Find the MedDRA ID for "Hepatitis"
hepatitis_id <- merged_db$onsides$vocab_meddra_adverse_effect %>%
  filter(meddra_name == "Hepatitis") %>%
  pull(meddra_id)

# 2. Find all drug ingredients linked to this effect in the high_confidence table
drug_ids_causing_hepatitis <- merged_db$onsides$high_confidence_enriched %>%
  filter(effect_meddra_id == hepatitis_id) %>%
  pull(drugbank_id) %>%
  na.omit() %>%
  unique()

# 3. Look up the targets for these DrugBank IDs
targets_of_interest <- merged_db$targets %>%
  filter(parent_key %in% drug_ids_causing_hepatitis) %>%
  select(drug_id = parent_key, target_name = name, gene_name)

head(targets_of_interest)

## End(Not run)
```

`merge_drugbank_twosides` *Merge a DrugBank dvobject with a TWOSIDES dvobject*

Description

Integrates drug-drug interaction data from TWOSIDES with the rich mechanistic information from DrugBank. This function is chainable and can accept a raw DrugBank object or an already-merged dvobject.

Usage

```
merge_drugbank_twosides(db_object, twosides_db)
```

Arguments

`db_object`	A dvobject from ‘parseDrugBank()‘ or an existing merged dvobject.
`twosides_db`	A dvobject from ‘parseTWOSIDES()‘.

Value

A new, nested dvobject with the TWOSIDES data added.

See Also

Other mergers: `merge_drugbank_onsides()`

parseDrugBank *parseDrugBank*

Description

parses given DrugBank XML database into a dvobject. dvobject is a list of data.frames in which each data.frame represents a part of parsed data (i.e drugs, prices, carriers, ...)

Usage

```
parseDrugBank(
  db_path,
  drug_options = NULL,
  parse_salts = FALSE,
  parse_products = FALSE,
  references_options = NULL,
  cett_options = NULL
)
```

Arguments

db_path	**string**, full path for the **DrugBank** xml or zip file.
drug_options	**character vector**, list of sub drug related nodes names options to parse (default = NULL). Check drug_node_options() for all available options. If its value is 'NULL' ONLY 'drug_general_information' will be placed in the returned dvobject.
parse_salts	**boolean**, parse salts info (default = FALSE)
parse_products	**boolean**, parse products info (default = FALSE)
references_options	**character vector**, list of sub references related nodes names options to parse (default = NULL). Check references_node_options() for all available options.
cett_options	**character vector**, list of sub cett related nodes names options to parse (default = NULL). Check cett_nodes_options() for all available options.

Value

dvobject

See Also

Other parsers: cett_nodes_options(), drug_node_options(), parseOnSIDES(), parseTWOSIDES(), references_node_options()

parseOnSIDES *Parse the OnSIDES Core Relational Database*

Description

Parses the core relational tables from the OnSIDES database.

Usage

```
parseOnSIDES(
  dataDir,
  include_high_confidence = TRUE,
  db_version = NULL,
  db_exported_date = NULL
)
```

Arguments

dataDir	A string specifying the path to the directory containing the OnSIDES CSV files.
include_high_confidence	Logical. If TRUE (the default), the function will also parse the 'high_confidence.csv' file, which is a pre-aggregated summary of ingredient-to-effect relationships. If the file is not found, a warning is issued.
db_version	used onside version (default = NULL)
db_exported_date	used onside release date (default = NULL)

Value

dvobject

See Also

Other parsers: cett_nodes_options(), drug_node_options(), parseDrugBank(), parseTWOSIDES(), references_node_options()

Other parsers: cett_nodes_options(), drug_node_options(), parseDrugBank(), parseTWOSIDES(), references_node_options()

`parseTWOSIDES` *Parse the TWOSIDES Drug-Drug Interaction Database*

Description

Reads the TWOSIDES data file, which contains adverse event data for pairs of drugs taken concurrently (N=2 interactions).

Usage

```
parseTWOSIDES(twosides_file_path, db_version = NULL, db_exported_date = NULL)
```

Arguments

`twosides_file_path`	Path to the TWOSIDES data file (e.g., 'TWOSIDES.csv.gz').
`db_version`	used twoside version (default = NULL)
`db_exported_date`	used twoside release date (default = NULL)

Details

TWOSIDES is a database of drug-drug interaction safety signals mined from the FDA's Adverse Event Reporting System using the same approach as is used to generate OffSIDES.

Database fields as follow:

drug_1_rxnorn_id RxNORM identifier for drug 1

drug_1_concept_name RxNORM name string for drug 1

drug_2_rxnorm_id RxNORM identifier for drug 2

drug_2_concept_name RxNORM name string for drug 3

condition_meddra_id MedDRA identifier for the side effect

condition_concpet_name MedDRA name string for the side effect

A The number of reports for the pair of drugs that report the side effect

B The number of reports for the pair of drugs that do not report the side effect

C The number of reports for other PSM matched drugs (including perhaps the single versions of drug 1 or drug 2) that report the side effect

D The number of reports for other PSM matched drugs and other side effects

PRR Proportional reporting ratio, PRR=(A/(A+B))/(C/(C+D))

PRR_error Error estimate of the PRR

mean_reporting_frequency Proportion of reports for the drug that report the side effect, A/(A+B)

Value

A dvobject of class 'TWOSIDESDB' containing the 'drug_drug_interactions' data frame and associated metadata.

See Also

Other parsers: `cett_nodes_options()`, `drug_node_options()`, `parseDrugBank()`, `parseOnSIDES()`, `references_node_options()`

---

references_node_options

returns references node valid options.

Description

returns references node valid options.

Usage

```
references_node_options()
```

Value

list of references valid options

See Also

Other parsers: `cett_nodes_options()`, `drug_node_options()`, `parseDrugBank()`, `parseOnSIDES()`, `parseTWOSIDES()`

---

show_dvobject_metadata

Display dvobject Metadata

Description

Displays information about passed dbobject object including basic info, database metadata, and all data.frames contained within nested lists.

Usage

```
show_dvobject_metadata(obj, return_df = FALSE)
```

Arguments

obj	A dvobject
return_df	Logical. If TRUE, returns metadata data.frame without printing. Default is FALSE.

Value

Invisibly returns a data.frame containing dvobject metadata

See Also

Other utility: add_database_info(), subset_drugbank_dvobject(), subset_onsides_dvobject()

Examples

```
## Not run:
display_merged_db_attrs(drugbank)
metadata <- display_merged_db_attrs(drugbank, return_df = TRUE)

## End(Not run)
```

---

subset_drugbank_dvobject

Subset a DrugBank dvobject by a vector of DrugBank IDs

Description

Subset a DrugBank dvobject by a vector of DrugBank IDs

Usage

```
subset_drugbank_dvobject(dvobject, drug_ids)
```

Arguments

dvobject	The dvobject from ‘parseDrugBank()‘.
drug_ids	A character vector of ‘drugbank_id‘ values to keep.

Details

Intelligently filters a DrugBank dvobject to retain only the data associated with a specified list of drugbank_ids. It correctly handles the deep, multi-level nested structure of the entire object, including the complex relationships within the ‘cett‘ list.

Value

A new, smaller dvobject with the same structure and attributes.

See Also

Other utility: add_database_info(), show_dvobject_metadata(), subset_onsides_dvobject()

Examples

```
## Not run:
library(dbparser)
one_drug <- subset_drugbank_dvobject(dvobject = dbdataset::drugbank,
                                     drug_ids = "DB00001")

## End(Not run)
```

---

subset_onsides_dvobject

Subset an OnSIDES dvobject by a vector of RxNorm Ingredient IDs (Schema-Aware)

Description

Intelligently filters an OnSIDES dvobject by cascading filters through the relational tables, ensuring the final subset is self-consistent.

Usage

```
subset_onsides_dvobject(dvobject, ingredient_ids)
```

Arguments

dvobject	A dvobject from 'parseOnSIDES()'.
ingredient_ids	A character vector of RxNorm CUIs (ingredients) to keep.

Value

A new, smaller dvobject with the same structure.

See Also

Other utility: add_database_info(), show_dvobject_metadata(), subset_drugbank_dvobject()

Index