Retrieve clinical trial information

Ralf Herold

2022-03-20

Get started

Attach package `ctrdata`

```
library(ctrdata)
citation("ctrdata")
```

Remember to respect the registers' terms and conditions (see `ctrOpenSearchPagesInBrowser(copyright = TRUE)`). Please cite this package in any publication as follows: Ralf Herold (2022). ctrdata: Retrieve and Analyze Clinical Trials in Public Registers. R package version 1.9.0. https://cran.r-project.org/package=ctrdata

Open register's advanced search page in browser

These functions open the browser, where the user can start searching for trials of interest.

```
# Please review and respect register copyrights:
ctrOpenSearchPagesInBrowser(
  copyright = TRUE
)
# Open browser with example search:
ctrOpenSearchPagesInBrowser(
  url = "cancer&age=under-18",
  register = "EUCTR"
)
```

Adjust search parameters and execute search in browser

Refine the search until the trials of interest are listed in the browser. The total number of trials that can be retrieved with package `ctrdata` is intentionally limited to queries with at most 10000 result records.

Copy address from browser address bar to clipboard

Use functions or keyboard shortcuts according to the operating system.

Get address from clipboard

The next steps are executed in the R environment:

```
q <- ctrGetQueryUrl()
# * Using clipboard content as register query URL: https://www.clinicaltrialsregister.eu/ctr-search/sea
# * Found search query from EUCTR: query=cancer&age=under-18&status=completed&phase=phase-one

q
#                                                  query-term  query-register
# 1 query=cancer&age=under-18&status=completed&phase=phase-one           EUCTR

# To check, this opens a browser with the query
ctrOpenSearchPagesInBrowser(url = q)
```

Retrieve protocol-related information, transform, save to database, check

```
# Count number of trial records
ctrLoadQueryIntoDb(
  queryterm = q,
  only.count = TRUE
)$n
# (1/3) Checking trials in EUCTR:
# Retrieved overview, multiple records of 71 trial(s) from 4 page(s) to be downloaded
# [1] 71

# Connect to a database and chose a table / collection
db <- nodbi::src_sqlite(
  dbname = "sqlite_file.sql",
  collection = "test"
)

# Retrieve records, download into database
ctrLoadQueryIntoDb(
  queryterm = q,
  con = db
)
# Checking helper binaries: done
# (1/3) Checking trials in EUCTR:
# Retrieved overview, multiple records of 71 trial(s) from 4 page(s) to be downloaded
# Downloading trials (4 pages in parallel)...
# Note: register server cannot compress data, transfer takes longer, about 0.4s per trial
# Pages: 4 done, 0 ongoing
# (2/3) Converting to JSON, 281 records converted
# (3/3) Importing JSON records into database...
# = Imported or updated 281 records on 71 trial(s)
# Updated history ("meta-info" in "test")

# Show which queries have been downloaded into database
dbQueryHistory(con = db)
#       query-timestamp query-register query-records
# 1 2022-03-19 09:41:36          EUCTR           281
```

```
#                                                    query-term
# 1 query=cancer&age=under-18&status=completed&phase=phase-one
```

With a file-base SQLite database, this takes about 20 seconds for about 300 records, with most of the time needed for internet-retrieval with is slow from this register. Speed is higher with other registers, with using MongoDB and with memory-based SQLite.

Repeat and update a previous query

```
ctrLoadQueryIntoDb(
  querytoupdate = "last",
  con = db
)
```

Instead of "last", an integer number can be specified for `querytoupdate` that corresponds to the number when using `dbQueryHistory()`.

Depending on the register, an update (differential update) is possible or the original query is executed fully again.

Retrieve results

For EUCTR, result-related trial information has to be requested to be retrieved, because it will take longer to download and store. For CTGOV, any results are always included in the retrieval.

```
ctrLoadQueryIntoDb(
  queryterm = q,
  euctrresults = TRUE,
  con = db
)
# Checking helper binaries: done
# (1/3) Checking trials in EUCTR:
# Retrieved overview, multiple records of 71 trial(s) from 4 page(s) to be downloaded
# Downloading trials (4 pages in parallel)...
# Note: register server cannot compress data, transfer takes longer, about 0.2s per trial
# Pages: 4 done, 0 ongoing
# (2/3) Converting to JSON, 281 records converted
# (3/3) Importing JSON records into database...
# = Imported or updated 281 records on 71 trial(s)
# Retrieving results if available from EUCTR for 71 trials:
# (1/4) Downloading results (max. 10 trials in parallel):
# 71 downloaded, extracting x x x x x . x . PDF . . . x . . . x PDF . PDF . . PDF . x . .
# x . . x x . . x . . . . . . . . . . . . . x . . . . . . PDF . . . . . .
# PDF . . . . . PDF . PDF .
# (2/3) Converting to JSON, 57 records converted
# (3/4) Importing JSON into database...
# (4/4) Results history: not retrieved (euctrresultshistory = FALSE)
# = Imported or updated results for 57 trials
# Updated history ("meta-info" in "test")
```

With a file-base SQLite database, this takes about 4 minutes for about 300 records, with most of the time needed for merging result- and protocol-related information in SQLite; this is much faster with MongoDB and PostgreSQL.

The download or presence of results is not recorded in `dbQueryHistory()` because the availability of results increases over time.

Add information from another register

The same collection can be used to store (and analyse) trial information from different registers. At the moment, `ctrdata` supports three registers: https://ClinicalTrials.Gov/, https://ClinicalTrialsRegister.EU/, https://www.ISRCTN.com/ and https://euclinicaltrials.eu/. Example:

```
ctrLoadQueryIntoDb(
  queryterm = "https://clinicaltrials.gov/ct2/results?cond=neuroblastoma&recrs=e&age=0&intr=Drug",
  con = db
)
# * Found search query from CTGOV: cond=neuroblastoma&recrs=e&age=0&intr=Drug
# Checking helper binaries: done
# (1/3) Checking trials in CTGOV:
# Retrieved overview, records of 200 trial(s) are to be downloaded (estimate: 1.6 MB)
# Downloading: 1.4 MB
# (2/3) Converting to JSON, 200 records converted
# (3/3) Importing JSON records into database...
# = Imported or updated 175 trial(s)
# Updated history ("meta-info" in "test")

# Warning message:
# [...]/ctrDATA15bc13c33351a/ctgov_trials_7.ndjson: Error : C stack usage  8342562 is too close to the
```

With a file-base SQLite database, this takes about 10 seconds for about 200 records.

Note that in this example, a warning message is issued from importing an NDJSON file with trial records. The warning arises from the high level of complexity of some of the XML content of some of the trial records. The issue can be resolved by increasing in the operating system the stack size available to R, see: https://github.com/rfhb/ctrdata/issues/22

Add personal annotations

When downloading trial information, the user can specify an annotation to all records that are downloaded. By default, annotations are accumulated if trial records are loaded again or updated; alternatively, annotations can be replaced.

Annotations are useful for analyses, for example to specially identify subsets of records in the database.

```
ctrLoadQueryIntoDb(
  queryterm = "https://clinicaltrials.gov/ct2/results?cond=neuroblastoma&recrs=e&age=0&intr=Drug&cntry=
  annotation.text = "site_DE ",
  annotation.mode = "append",
  con = db
)
# * Found search query from CTGOV: cond=neuroblastoma&recrs=e&age=0&intr=Drug&cntry=DE
# Checking helper binaries: done
```

```
# (1/3) Checking trials in CTGOV:
# Retrieved overview, records of 11 trial(s) are to be downloaded (estimate: 0.088 MB)
# Downloading: 69 kB
# (2/3) Converting to JSON, 11 records converted
# (3/3) Importing JSON records into database...
# = Imported or updated 11 trial(s)
# = Annotated retrieved records (11 records)
# Updated history ("meta-info" in "test")
```

Find synonyms of active substance names

Not all registers automatically expand search terms to include alternative terms, such as codes and other names of active substances. To obtain a character vector of synonyms for any active substance name, use:

```
ctrFindActiveSubstanceSynonyms(
  activesubstance = "imatinib"
)
# [1] "imatinib"  "gleevec" "sti 571" "glivec" "CGP 57148" "st1571"
```

These names can then be used in queries in any register.

Using a MongoDB database

This example works with a free service here. Note that the user name and password need to be encoded. The format of the connection string is documented at https://docs.mongodb.com/manual/reference/connection-string/.

For recommended databases, see vignette Install R package ctrdata.

```
# Specify base uri for remote MongoDB server,
#  as part of the encoded connection string
db <- nodbi::src_mongo(
  # Note: this provides read-only access
  url = "mongodb+srv://DWbJ7Wh:bdTHh5cS@cluster0-b9wpw.mongodb.net",
  db = "dbperm",
  collection = "dbperm")

# Since the above access is read-only,
# just obtain fields of interest:
dbGetFieldsIntoDf(
  fields = c("a2_eudract_number",
             "e71_human_pharmacology_phase_i"),
  con = db)
#                   _id a2_eudract_number e71_human_pharmacology_phase_i
# 1 2010-024264-18-3RD    2010-024264-18                            TRUE
# 2  2010-024264-18-AT    2010-024264-18                            TRUE
# 3  2010-024264-18-DE    2010-024264-18                            TRUE
# 4  2010-024264-18-GB    2010-024264-18                            TRUE
# 5  2010-024264-18-IT    2010-024264-18                            TRUE
# 6  2010-024264-18-NL    2010-024264-18                            TRUE
```