Deduplication of bibliographic records with ASySD in R

2025-07-01

This is a quarto document that contains both human language and R code. It works almost exactly as an R markdown file (.Rmd). click here for more info.

The structure of this R project is the same as the Utrecht University (UU) simple R project.

RIS files

For this deduplication challenges, all files are made available as Research Information Systems Incorporated (RIS) files, which can be read by the `load_search` function from the `ASySD` package. The RIS files are available in the `data/raw` folder. The four datasets to be deduplicated were presumably obtained from the following databases:

- **Embase**
- **Lens**
- **OpenAlex**
- **Scopus**

We will use ASySD to load the datasets by using the `load_search` function for a RIS file.

Embase

```
# Load embase data
embase_raw <- load_search(
  path = paste0(inputfolder, "/5_ASReviewSummSchool_Embase.ris"),
  method = "ris"
  )
```

```r
# Examine the names of the loaded dataset
embase_raw %>% names
```

```
 [1] "database"        "source_type"     "language"        "author"         
 [5] "address"         "year"            "title"           "journal"        
 [9] "volume"          "issue"           "pages"           "abstract"       
[13] "keywords"        "doi"             "issn"            "url"            
[17] "C5"              "L2"              "LK"              "M3"             
[21] "U2"              "U3"              "U4"              "place_published"
[25] "number"          "record_id"       "isbn"            "label"          
[29] "source"         
```

There are a total of 5 records in the embase dataset.

Since ASySD expects the data to have a rather strict structure, further processing is necessary. The columns that ASySD expects are:

Name	Definition
author	The author(s) of the publication
year	The year the publication was published
journal	The name of the journal in which the publication appeared
doi	The Digital Object Identifier (DOI) assigned to the publication
title	The title of the publication
pages	The page numbers of the publication
volume	The volume number of the publication (if applicable)
number	The issue number of the publication (if applicable)
abstract	Abstract of publication
record_id	A unique identifier for the publication. If this is not obtained from the citation file, ASySD will genereate an id for each citation based on row numbers.
isbn	The International Standard Book Number (ISBN) assigned to the publication (if applicable). If unavailable, the International Standard Serial Number can be used here instead (ISSN).
label (optional)	A label or tag assigned to the publication (if applicable) - for example, **new search** or **old search**
source (optional)	The source or database from which the publication was obtained - for example **wos**, **embase**, **pubmed**, **scopus**

This table was extracted and exactly reproduced from ASySD GitHub site.

We thus need to select the columns that ASySD expects and rename them accordingly.

```
# Columns to select
columns <- c("record_id", "author", "year", "journal", "doi", "title", "pages",
             "volume", "number", "abstract", "isbn", "label", "source")
```

```
embase <- embase_raw %>%
  mutate(
    record_id = record_id, # None available in this dataset,
                           # empty column created by load_search
    author = author,     # Correct, pattern = "Last, F. S. and"
    year = year,         # Correct
    journal = journal,   # Correct
    doi = doi,           # Correct
    title = title,       # Correct
    pages = pages,       # Correct, note separated by "-" with no spaces.
    volume = volume,     # Correct
    number = issue,      # Called issue in original dataset
    abstract = abstract, # Correct
    isbn = issn,         # Called issn in original dataset, may have >1 separated
                         # by " and " with spaces.
    label = label,       # Empty column created by load_search
    source = database    # Called database in original dataset
    ) %>%
  select(all_of(columns)) %>%
  mutate_if(is.character, ~na_if(., "")) # Replace empty strings with NA
```

Scopus

```
# Load scopus data
scopus_raw <- load_search(
  path = paste0(inputfolder, "/25_ASreviewSummSchool_Scopus.ris"),
  method = "ris"
  )

# Examine the names of the loaded dataset
scopus_raw %>% names
```

```
 [1] "database"            "document_type"       "language"
 [4] "A2"                  "author"              "address"
 [7] "year"                "title"               "journal"
[10] "source_abbreviated"  "volume"              "pages"
[13] "abstract"            "keywords"            "doi"
```

```
[16] "issn"                 "url"                "publisher"
[19] "notes"                "ZZ"                 "source_type"
[22] "issue"                "article_number"     "supertaxa"
[25] "number"               "record_id"          "isbn"
[28] "label"                "source"
```

There are a total of 25 records in the scopus dataset.

```r
scopus <- scopus_raw %>%
  mutate(
    record_id = record_id, # None available in this dataset,
                           # empty column created by load_search
    author = author,      # Correct, pattern = "Last, F. S. and"
    year = year,          # Correct
    journal = journal,    # Correct
    doi = doi,            # Correct
    title = title,        # Correct
    pages = pages,        # Correct, note separated by "-" with no spaces.
    volume = volume,      # Correct
    number = issue,       # Called issue in original dataset
    abstract = abstract,  # Correct
    isbn = issn,          # Called issn in original dataset with
                          # "(ISSN)" or "(ISBN)" sting after,
                          # may have >1 separated by ";"
    label = label,        # Empty column created by load_search
    source = database     # Called database in original dataset
    ) %>%
  select(all_of(columns)) %>%
  mutate_if(is.character, ~na_if(., "")) # Replace empty strings with NA
```

OpenAlex

```r
# Load openalex data
openalex_raw <- load_search(
  path = paste0(inputfolder, "/2658_ASReviewSummerschool_OpenAlex.ris"),
  method = "ris"
  )

# Examine the names of the loaded dataset
openalex_raw %>% names
```

```
 [1] "date_generated" "source_type"    "language"       "author"
 [5] "year"           "title"          "journal"        "volume"
 [9] "issue"          "pages"          "abstract"       "keywords"
[13] "doi"            "issn"           "url"            "publisher"
[17] "C1"             "NG"             "BP"             "number"
[21] "record_id"      "isbn"           "label"          "source"
```

There are a total of 2665 records in the openalex dataset.

```
openalex <- openalex_raw %>%
  mutate(
    record_id = record_id, # None available in this dataset,
                           # empty column created by load_search
    author = author,      # Correct, pattern = "Last, First and"
    year = year,          # Correct
    journal = journal,    # Correct
    doi = doi,            # Correct
    title = title,        # Correct
    pages = pages,        # Correct, note separated by "-" with no spaces.
    volume = volume,      # Correct
    number = issue,       # Called issue in original dataset
    abstract = abstract,  # Correct
    isbn = issn,          # Called issn in original dataset, single issn.
    label = label,        # Empty column created by load_search
    source = "openalex"   # Not available, will call it openalex
    ) %>%
  select(all_of(columns)) %>%
  mutate_if(is.character, ~na_if(., "")) # Replace empty strings with NA
```

Lens

```
# Load lens data
lens_raw <- load_search(
  path = paste0(inputfolder, "/568-ASReviewSummerschool-LENS.ris"),
  method = "ris"
  )

# Examine the names of the loaded dataset
lens_raw %>% names
```

```
 [1] "date_generated" "source_type"    "author"         "address"
 [5] "year"           "title"          "journal"        "pages"
 [9] "abstract"       "doi"            "issn"           "url"
[13] "publisher"      "ID"             "L2"             "volume"
[17] "issue"          "keywords"       "L1"             "number"
[21] "record_id"      "isbn"           "label"          "source"
```

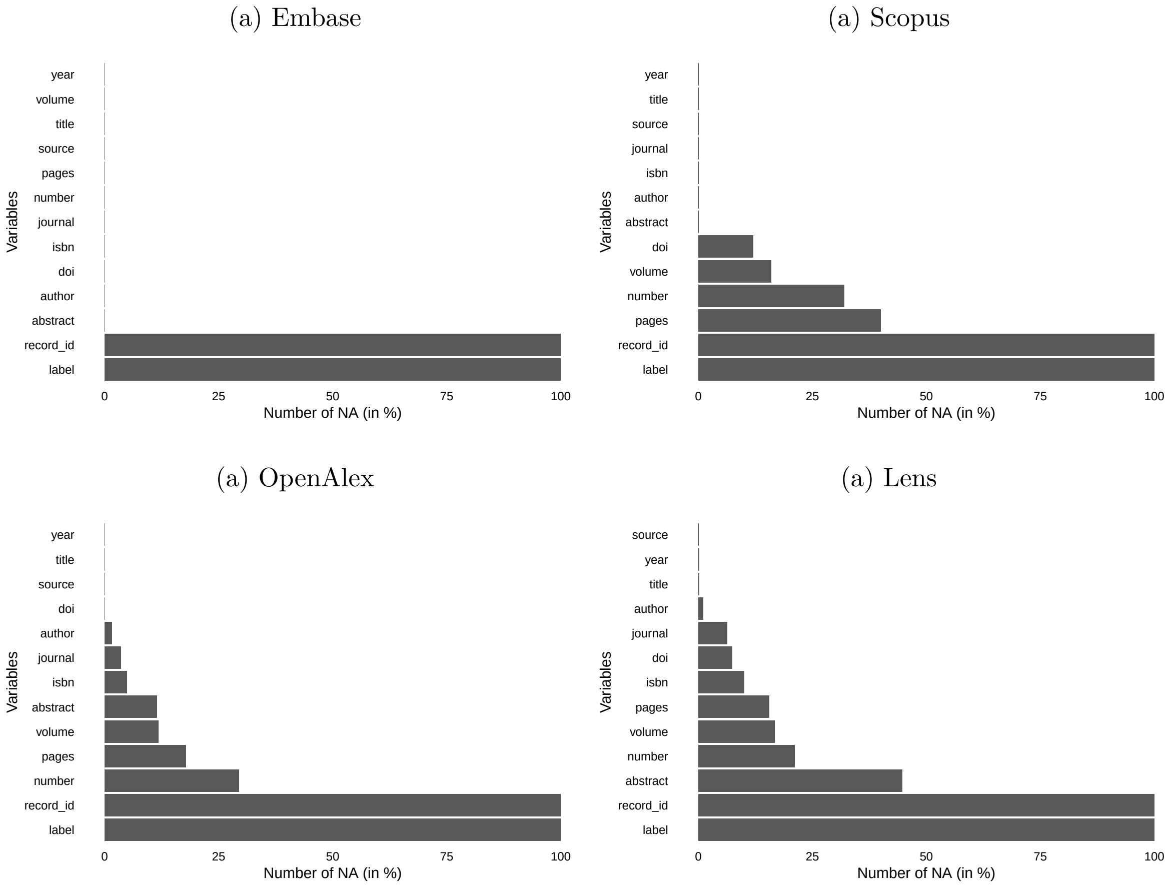
There are a total of 568 records in the lens dataset.

```
lens <- lens_raw %>%
  mutate(
    record_id = record_id, # None available in this dataset,
                           # empty column created by load_search
    author = author,      # Correct, pattern = "Last, First and"
    year = year,          # Correct
    journal = journal,    # Correct
    doi = doi,            # Correct
    title = title,        # Correct
    pages = pages,        # Correct, note separated by "-" with no spaces.
    volume = volume,      # Correct
    number = issue,       # Called issue in original dataset
    abstract = abstract,  # Correct, but html code is present.
    isbn = issn,          # Called issn in original dataset,
                          # may have >1 separated by " and " with spaces.
    label = label,        # Empty column created by load_search
    source = "lens"       # Not available, will call it lens
    ) %>%
  select(all_of(columns)) %>%
  mutate_if(is.character, ~na_if(., "")) # Replace empty strings with NA
```

Examine missing data

```
overview_na(embase)
overview_na(scopus)
overview_na(openalex)
overview_na(lens)
```

```
# Bind all datasets
records <- bind_rows(
  embase,
```



(a) Embase

(a) Scopus

(a) OpenAlex

(a) Lens

Figure 1. Missing data per database

```
  scopus,
  openalex,
  lens
  )
```

There are a total of 3263 records. These will be deduplicated using the Automated Systematic Search Deduplicator (ASySD).

```
# Deduplicate studies
deduplicated <- dedup_citations(
  records,
  manual_dedup =  TRUE,
  show_unknown_tags = FALSE,
  user_input = 1
  )

# If only journal articles, removing doi exact matches could be appropriate
# this will remove many manual duplicates already, useful for large datasets.
records_unique <- deduplicated$unique
record_manual_dedup <- deduplicated$manual_dedup %>%
  mutate(
    result = case_when(
      doi >0.9999 ~ TRUE,
      TRUE ~ NA
    )
  )
```

```
# This will open a Shiny app to manually deduplicate the records.
true_dups <- manual_dedup_shiny(record_manual_dedup)

# Saved as a temporary file to prevent any progress lost.
saveRDS(true_dups, file = paste0(tempfolder, "/true_duplicates.rds"))
```

```
# Reload the true duplicates from the temporary file.
true_dups <- readRDS(paste0(tempfolder, "/true_duplicates.rds"))

# Incorporate manual decisions into the final dataset.
final_dedup <- dedup_citations_add_manual(records_unique, additional_pairs = true_dups)
```

```
write_citations(
  final_dedup,
  type = "csv",
  filename = paste0(psfolder, "/deduplicated_final.csv")
  )
```

Final thoughts

After deduplication, there were a total of 3137 studies. Thus, the final number of duplicated records was: 126.

However, the data had remaining unfixed issues such as different name patterns and issn, so it would be good to examine if fixing this changes the results.

Overall, ASySD is quite useful but may require quite a lot of coding and data preparation to get reliable results.

Session Information

```r
# remove clutter
session <- sessionInfo()
session$BLAS <- NULL
session$LAPACK <- NULL
session$loadedOnly <- NULL

session
```

```
R version 4.4.0 (2024-04-24 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows 11 x64 (build 26100)

Matrix products: default


locale:
[1] LC_COLLATE=Dutch_Netherlands.utf8  LC_CTYPE=Dutch_Netherlands.utf8
[3] LC_MONETARY=Dutch_Netherlands.utf8 LC_NUMERIC=C
[5] LC_TIME=Dutch_Netherlands.utf8

time zone: Europe/Amsterdam
tzcode source: internal

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] ASySD_0.4.1      report_0.6.1     gt_0.11.0        overviewR_0.0.13
 [5] lubridate_1.9.3  forcats_1.0.0    stringr_1.5.1    dplyr_1.1.4
 [9] purrr_1.0.2      readr_2.1.5      tidyr_1.3.1      tibble_3.2.1
[13] ggplot2_3.5.1    tidyverse_2.0.0  devtools_2.4.5   usethis_3.0.0
[17] pacman_0.5.1
```

Package References

```
report::cite_packages(session)
```

- Grolemund G, Wickham H (2011). "Dates and Times Made Easy with lubridate." *Journal of Statistical Software*, *40*(3), 1-25. https://www.jstatsoft.org/v40/i03/.
- Hair K, Bahor Z, Macleod M, Liao J, Sena ES (2021). "The Automated Systematic Search Deduplicator (ASySD): a rapid, open-source, interoperable tool to remove duplicate citations in biomedical systematic reviews." *bioRxiv*. doi:10.1101/2021.05.04.442412 https://doi.org/10.1101/2021.05.04.442412.
- Iannone R, Cheng J, Schloerke B, Hughes E, Lauer A, Seo J, Brevoort K, Roy O (2024). *gt: Easily Create Presentation-Ready Display Tables*. R package version 0.11.0, https://CRAN.R-project.org/package=gt.
- Makowski D, Lüdecke D, Patil I, Thériault R, Ben-Shachar M, Wiernik B (2023). "Automated Results Reporting as a Practical Tool to Improve Reproducibility and Methodological Best Practices Adoption." *CRAN*. https://easystats.github.io/report/.
- Meyer C, Hammerschmidt D (2023). *overviewR: Easily Extracting Information About Your Data*. R package version 0.0.13, https://CRAN.R-project.org/package=overviewR.
- Müller K, Wickham H (2023). *tibble: Simple Data Frames*. R package version 3.2.1, https://CRAN.R-project.org/package=tibble.
- R Core Team (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.
- Rinker TW, Kurkiewicz D (2018). *pacman: Package Management for R*. version 0.5.0, http://github.com/trinker/pacman.
- Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4, https://ggplot2.tidyverse.org.
- Wickham H (2023). *forcats: Tools for Working with Categorical Variables (Factors)*. R package version 1.0.0, https://CRAN.R-project.org/package=forcats.
- Wickham H (2023). *stringr: Simple, Consistent Wrappers for Common String Operations*. R package version 1.5.1, https://CRAN.R-project.org/package=stringr.
- Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the tidyverse." *Journal of Open Source Software*, *4*(43), 1686. doi:10.21105/joss.01686 https://doi.org/10.21105/joss.01686.
- Wickham H, Bryan J, Barrett M, Teucher A (2024). *usethis: Automate Package and Project Setup*. R package version 3.0.0, https://CRAN.R-project.org/package=usethis.
- Wickham H, François R, Henry L, Müller K, Vaughan D (2023). *dplyr: A Grammar of Data Manipulation*. R package version 1.1.4, https://CRAN.R-project.org/package=dplyr.
- Wickham H, Henry L (2023). *purrr: Functional Programming Tools*. R package version 1.0.2, https://CRAN.R-project.org/package=purrr.

- Wickham H, Hester J, Bryan J (2024). *readr: Read Rectangular Text Data*. R package version 2.1.5, https://CRAN.R-project.org/package=readr.
- Wickham H, Hester J, Chang W, Bryan J (2022). *devtools: Tools to Make Developing R Packages Easier*. R package version 2.4.5, https://CRAN.R-project.org/package=devtools.
- Wickham H, Vaughan D, Girlich M (2024). *tidyr: Tidy Messy Data*. R package version 1.3.1, https://CRAN.R-project.org/package=tidyr.