

Package ‘TaxResolveR’

September 21, 2026

Title Taxonomic Name Resolution and Validation Tools

Version 0.1.0

Description Provides reproducible tools for cleaning, parsing, classifying, standardising, validating and resolving scientific names in ecological and biodiversity datasets. Taxonomic matches can be assessed for match quality and taxonomic status, records requiring manual review can be identified, and resolution results can be summarised, reported and exported. Taxonomic name resolution can use the 'GBIF' species matching service and the GBIF Backbone Taxonomy described by GBIF Secretariat (2023) [<doi:10.15468/39omei>](https://doi.org/10.15468/39omei).

URL <https://github.com/prdelosriosescalante/TaxResolveR>

BugReports <https://github.com/prdelosriosescalante/TaxResolveR/issues>

License GPL-3

Encoding UTF-8

Language en-GB

RoxygenNote 8.0.0

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/spelling/language en-GB

VignetteBuilder knitr

Imports httr2

NeedsCompilation no

Author Patricio R. De los Ríos-Escalante [aut, cre] (ORCID:
 [<https://orcid.org/0000-0001-5056-7003>](https://orcid.org/0000-0001-5056-7003))

Maintainer Patricio R. De los Ríos-Escalante <patricio.delosrios.e@mail.pucv.cl>

Repository CRAN

Date/Publication 2026-09-21 18:10:02 UTC

Contents

assess_taxonomic_match	2
classify_scientific_names	3
clean_scientific_names	5
export_taxonomic_results	6
flag_taxonomic_review	7
parse_scientific_names	9
prepare_taxonomic_queries	10
resolve_taxonomy	11
standardize_scientific_names	12
summarize_taxonomic_resolution	13
taxonomic_resolution_report	15
taxresolve	16
validate_species_names	17
Index	18

assess_taxonomic_match
<i>Assess taxonomic match quality</i>

Description

Classifies taxonomic resolutions returned by `resolve_taxonomy()` according to resolution status, match quality, and taxonomic acceptance status.

Usage

```
assess_taxonomic_match(x)
```

Arguments

- x A data frame containing taxonomic resolution results, typically returned by `resolve_taxonomy()`. It must contain the columns `query_name`, `taxonomic_status`, `match_type`, and `resolution_success`.

Details

Resolution status distinguishes names that were successfully resolved, names that were queried but could not be resolved, and inputs that were not suitable for an external taxonomic query.

Match quality is assessed independently. Exact successful matches are classified as "high", successful non-exact matches as "moderate", successful matches lacking match-type information as "low", and queried but unresolved names as "unresolved".

Taxonomic acceptance status is assessed separately from match quality. Resolved names with taxonomic status "ACCEPTED" are classified as "accepted", synonyms as "not_accepted", and other non-missing taxonomic statuses as "uncertain". Queried but unresolved names are classified as "unresolved".

Inputs that were not queryable receive "not_queryable" as their resolution status and NA for match quality and taxonomic acceptance status.

Value

The input data frame with three additional character columns: resolution_status, match_quality, and accepted_status.

Examples

```
x <- data.frame(  
  query_name = c(  
    "Homo sapiens",  
    "Homo sapens",  
    "Unknown species",  
    NA_character_  
  ),  
  taxonomic_status = c(  
    "ACCEPTED",  
    "ACCEPTED",  
    NA_character_,  
    NA_character_  
  ),  
  match_type = c(  
    "EXACT",  
    "VARIANT",  
    NA_character_,  
    NA_character_  
  ),  
  resolution_success = c(  
    TRUE,  
    TRUE,  
    FALSE,  
    FALSE  
  ),  
  stringsAsFactors = FALSE  
)  
  
assess_taxonomic_match(x)
```

`classify_scientific_names`*Classify scientific names*

Description

Classifies scientific names according to their structural status after cleaning, parsing, and standardization.

Usage

```
classify_scientific_names(x)
```

Arguments

x Character vector containing scientific names.

Details

The function uses the existing TaxResolveR workflow to distinguish successfully parsed taxonomic names, genus-level placeholders, genus-only names, missing values, empty strings, and names that could not be parsed.

Structural classification does not constitute taxonomic validation. In particular, the status `valid_structure` indicates that the name has a syntactic structure recognised by TaxResolveR. It does not indicate that the taxon exists, that the name is currently accepted, or that it has been verified against an external taxonomic authority.

Value

A data.frame with one row per input name and the following columns:

original_name Original input supplied by the user.

clean_name Name after internal cleaning and formatting.

genus Parsed genus name.

specific_epithet Parsed specific epithet.

infraspecific_rank Explicit infraspecific rank, when present.

infraspecific_epithet Parsed infraspecific epithet.

rank_detected Taxonomic structure detected by the parser.

canonical_name Canonical name reconstructed from parsed components.

parse_success Logical value indicating successful syntactic parsing.

name_status Structural classification assigned to the input name.

Possible values of `name_status` are:

valid_structure A species or supported infraspecific name was successfully parsed.

placeholder A genus-level placeholder using `sp.` was detected.

genus_only A genus name without a specific epithet was detected.

missing The original input was a missing value.

empty The input was empty or contained only whitespace after cleaning.

unparsed A non-empty input could not be interpreted by the parser.

Examples

```
classify_scientific_names(c(
  "Homo sapiens",
  "Chilina sp.",
  "Chilina",
  "Parastacus brasiliensis subsp. promatensis",
  NA_character_,
  "",
  "12345"
))
```

`clean_scientific_names`*Clean scientific names*

Description

Standardises the format of scientific names before taxonomic validation or external database queries.

Usage

```
clean_scientific_names(x)
```

Arguments

`x` Character vector containing scientific names.

Details

Missing values are preserved as missing values.

Value

A cleaned character vector.

Examples

```
clean_scientific_names(c(
  " homo sapiens ",
  "Panthera\tleo",
  "Chilina sp .",
  NA_character_
))
```

export_taxonomic_results

Export Taxonomic Resolution Results

Description

Exports the results of a TaxResolveR taxonomic resolution workflow to a set of standardised CSV files.

Usage

```
export_taxonomic_results(x, path, overwrite = FALSE)
```

Arguments

x	A data frame containing taxonomic resolution results. Typically, this is the object returned by <code>taxresolve()</code> . The data frame must contain the columns <code>resolution_status</code> , <code>match_quality</code> , <code>accepted_status</code> , <code>review_required</code> , and <code>review_reason</code> .
path	A single non-empty character string specifying the directory where the CSV files will be written. If the directory does not exist, it is created recursively.
overwrite	A single non-missing logical value indicating whether existing TaxResolveR output files should be replaced. The default is FALSE.

Details

The function writes the complete input data frame together with summary tables describing overall taxonomic resolution, resolution status, match quality, and reasons for manual taxonomic review.

Existing output files are not overwritten unless `overwrite = TRUE`. Files unrelated to TaxResolveR are never removed or modified.

Value

Invisibly returns a named character vector containing the normalised paths of the five exported CSV files:

taxonomic_results Complete taxonomic resolution results.

taxonomic_summary Overall taxonomic resolution summary.

resolution_status Counts and proportions by resolution status.

match_quality Counts and proportions by match quality.

review_reasons Counts and proportions by manual review reason.

Examples

```
x <- data.frame(
  resolution_status = c(
    "resolved",
    "resolved",
    "not_queryable"
  ),
  match_quality = c(
    "high",
    "moderate",
    NA
  ),
  accepted_status = c(
    "accepted",
    "accepted",
    NA
  ),
  review_required = c(
    FALSE,
    TRUE,
    TRUE
  ),
  review_reason = c(
    NA,
    "non_exact_match",
    "not_queryable"
  ),
  stringsAsFactors = FALSE
)

output_dir <- tempfile(
  "TaxResolveR_example_"
)

paths <- export_taxonomic_results(
  x,
  path = output_dir
)

names(paths)

unlink(
  output_dir,
  recursive = TRUE,
  force = TRUE
)
```

Description

Identifies taxonomic resolution records that should be reviewed manually based on resolution status, match quality, and taxonomic acceptance status.

Usage

```
flag_taxonomic_review(x)
```

Arguments

x A data frame containing taxonomic assessment results, typically returned by [assess_taxonomic_match\(\)](#).

Details

Records representing resolved, high-quality, accepted taxonomic matches are not flagged. Non-exact matches, low-information matches, unresolved names, non-queryable inputs, synonyms, uncertain taxonomic statuses, and records with insufficient assessment information are flagged for review.

Value

The input data frame with two additional columns: `review_required`, a logical vector indicating whether manual review is recommended, and `review_reason`, a character vector describing the primary reason for review.

Examples

```
x <- data.frame(
  resolution_status = c(
    "resolved",
    "resolved",
    "resolved",
    "unresolved"
  ),
  match_quality = c(
    "high",
    "moderate",
    "high",
    "unresolved"
  ),
  accepted_status = c(
    "accepted",
    "accepted",
    "not_accepted",
    "unresolved"
  ),
  stringsAsFactors = FALSE
)

flag_taxonomic_review(x)
```

`parse_scientific_names`*Parse scientific names*

Description

Parses scientific names into their main taxonomic components.

Usage

```
parse_scientific_names(x)
```

Arguments

`x` Character vector containing scientific names.

Details

The function recognises genus names, binomial species names, simple infraspecific names, explicit subspecies names, genus-level placeholders using "sp.", and missing values.

Value

A data.frame containing the original name, genus, specific epithet, infraspecific rank, infraspecific epithet, and detected rank.

Examples

```
parse_scientific_names(c(
  "Homo sapiens",
  "Chilina sp.",
  "Chilina",
  "Parastacus brasiliensis promatensis",
  "Parastacus brasiliensis subsp. promatensis",
  NA
))
```

```
prepare_taxonomic_queries
```

Prepare taxonomic queries

Description

Prepares scientific names for subsequent queries to external taxonomic authorities.

Usage

```
prepare_taxonomic_queries(x)
```

Arguments

x Character vector containing scientific names.

Details

The function uses the structural classification generated by `classify_scientific_names()` to determine the appropriate name to submit to an external taxonomic authority.

Species and supported infraspecific names are queried using their canonical names. Genus-only names are queried using the genus. Genus-level placeholders such as "Chilina sp." are reduced to the corresponding genus name.

Missing, empty, and unparsed inputs are not considered suitable for external taxonomic queries and therefore receive NA as their query value.

This function prepares taxonomic queries only. It does not communicate with GBIF, WoRMS, Catalogue of Life, or any other external taxonomic authority.

The function distinguishes between the name originally supplied by the user and the name prepared for an external query. For example, "Chilina sp." is preserved as the original name, while its query is prepared as "Chilina". This avoids treating an identification at genus level as if it represented a species-level scientific name.

A value of TRUE in `query_ready` means only that TaxResolveR was able to construct a syntactically suitable query. It does not imply that the taxon exists, that the name is taxonomically accepted, or that an external authority will return a match.

Value

A data.frame with one row per input name. The output contains all columns generated by `classify_scientific_names()`, together with:

query_name Character value containing the scientific name that should be submitted to an external taxonomic authority. Unsuitable inputs receive NA.

query_ready Logical value indicating whether a suitable taxonomic query could be generated.

Examples

```
prepare_taxonomic_queries(c(
  "Homo sapiens",
  "Chilina sp.",
  "Chilina",
  "Parastacus brasiliensis subsp. promatensis",
  NA_character_,
  "",
  "12345"
))
```

resolve_taxonomy	<i>Resolve scientific names against a taxonomic source</i>
------------------	--

Description

Resolves scientific names against a supported external taxonomic source and returns the results using the standard TaxResolveR taxonomic result structure.

Usage

```
resolve_taxonomy(x, source = "gbif")
```

Arguments

x	Character vector containing scientific names.
source	Character string specifying the taxonomic source. Currently, "gbif" is supported.

Details

Names are first prepared using the TaxResolveR scientific-name workflow. Only names considered suitable for external taxonomic queries are submitted to the selected backend.

Duplicate query names are resolved only once by the corresponding backend and are subsequently restored to their original positions.

Value

A data frame containing one taxonomic resolution result for each input value.

Examples

```

resolve_taxonomy(
  c(
    "Homo sapiens",
    "Chilina",
    "Parastacus brasiliensis"
  )
)

resolve_taxonomy(
  c(
    " homo sapiens ",
    " chilina ",
    NA_character_
  ),
  source = "gbif"
)

```

```
standardize_scientific_names
```

```
Standardize scientific names
```

Description

Cleans and parses scientific names into a standardized taxonomic representation suitable for subsequent taxonomic resolution.

Usage

```
standardize_scientific_names(x)
```

Arguments

x Character vector containing scientific names.

Details

The function preserves the original input, generates a cleaned version of each name, extracts supported taxonomic components, reconstructs a canonical scientific name, and reports whether the name was successfully parsed.

Supported structures currently include genus names, genus-level placeholders using `sp.`, binomial species names, simple trinomial names, and explicit infraspecific ranks using `subsp.`, `var.`, and `f.`.

The `parse_success` field indicates whether `TaxResolveR` successfully interpreted the syntactic structure of the name. It does not indicate whether the taxon exists, whether the name is currently accepted, or whether it has been verified against an external taxonomic authority.

Value

A data.frame with one row per input name and the following columns:

original_name Original input supplied by the user.
clean_name Name after internal cleaning and formatting.
genus Parsed genus name.
specific_epithet Parsed specific epithet.
infraspecific_rank Explicit infraspecific rank, when present.
infraspecific_epithet Parsed infraspecific epithet.
rank_detected Taxonomic structure detected by the parser.
canonical_name Canonical name reconstructed from parsed components.
parse_success Logical value indicating successful syntactic parsing.

Examples

```
standardize_scientific_names(c(
  " homo sapiens ",
  "Chilina sp.",
  "Parastacus brasiliensis subsp. promatensis",
  "Brassica oleracea var. capitata",
  NA_character_
))
```

```
summarize_taxonomic_resolution
```

Summarize Taxonomic Resolution Results

Description

Produces a compact one-row summary of taxonomic resolution results.

The function counts the total number of names and the number of records assigned to the main resolution, match-quality, acceptance, and review categories.

Character category values are standardised internally by removing leading and trailing whitespace and converting text to lower case. Empty character values are treated as missing values.

Usage

```
summarize_taxonomic_resolution(x)
```

Arguments

x A data frame containing taxonomic resolution results. The object must contain the columns `resolution_status`, `match_quality`, `accepted_status`, and `review_required`.

Value

A one-row data frame with the following columns:

- `total_names`: total number of input records.
- `resolved`: number of resolved records.
- `unresolved`: number of unresolved records.
- `not_queryable`: number of non-queryable records.
- `high_quality`: number of high-quality matches.
- `moderate_quality`: number of moderate-quality matches.
- `low_quality`: number of low-quality matches.
- `accepted`: number of accepted taxonomic matches.
- `not_accepted`: number of non-accepted taxonomic matches.
- `review_required`: number of records requiring manual review.

Examples

```
x <- data.frame(
  resolution_status = c(
    "resolved",
    "unresolved",
    "not_queryable"
  ),
  match_quality = c(
    "high",
    "unresolved",
    NA_character_
  ),
  accepted_status = c(
    "accepted",
    "unresolved",
    NA_character_
  ),
  review_required = c(
    FALSE,
    TRUE,
    TRUE
  ),
  stringsAsFactors = FALSE
)

summarize_taxonomic_resolution(x)
```

taxonomic_resolution_report
Create a Taxonomic Resolution Report

Description

Creates a structured diagnostic report from the results of the TaxResolveR taxonomic resolution workflow.

Usage

```
taxonomic_resolution_report(x)
```

Arguments

x A data frame containing taxonomic resolution results. Typically, this is the object returned by `taxresolve()`. The data frame must contain the columns `resolution_status`, `match_quality`, `accepted_status`, `review_required`, and `review_reason`.

Details

The report summarises overall taxonomic resolution, resolution status, match quality, and reasons for manual taxonomic review. Counts and proportions are calculated using the total number of input records as the denominator.

Value

A named list containing four data frames:

total_summary Overall taxonomic resolution summary produced by `summarize_taxonomic_resolution()`.

resolution_status Counts and proportions for resolved, unresolved, and non-queryable records.

match_quality Counts and proportions for taxonomic match-quality categories.

review_reasons Counts and proportions for reasons requiring manual taxonomic review.

Examples

```
x <- data.frame(  
  resolution_status = c(  
    "resolved",  
    "resolved",  
    "not_queryable"  
  ),  
  match_quality = c(  
    "high",  
    "moderate",  
    NA  
  ),  
)
```

```

    accepted_status = c(
      "accepted",
      "accepted",
      NA
    ),
    review_required = c(
      FALSE,
      TRUE,
      TRUE
    ),
    review_reason = c(
      NA,
      "non_exact_match",
      "not_queryable"
    ),
    stringsAsFactors = FALSE
  )

  taxonomic_resolution_report(x)

```

taxresolve

Resolve and assess scientific names

Description

Provides a high-level interface for taxonomic name resolution in TaxResolveR. Scientific names are resolved against a supported taxonomic source, assessed for resolution and match quality, and flagged when manual taxonomic review is recommended.

Usage

```
taxresolve(x, source = "gbif")
```

Arguments

x	A character vector containing scientific names.
source	Character string specifying the taxonomic source. Currently, "gbif" is supported.

Details

This function combines [resolve_taxonomy\(\)](#), [assess_taxonomic_match\(\)](#), and [flag_taxonomic_review\(\)](#) into a single workflow.

Value

A data frame containing the taxonomic resolution returned by [resolve_taxonomy\(\)](#) together with the assessment columns produced by [assess_taxonomic_match\(\)](#) and the review columns produced by [flag_taxonomic_review\(\)](#).

Examples

```
taxresolve(  
  c(  
    "Homo sapiens",  
    "Homo sapens",  
    "Chilina sp."  
  )  
)
```

`validate_species_names`*Validate Species Names*

Description

Performs basic syntactic validation of species names, including missing values, empty names, repeated whitespace, and binomial structure.

Usage

```
validate_species_names(x)
```

Arguments

`x` Character vector containing scientific names.

Value

A data.frame with the original names and logical indicators for missing names, empty names, repeated whitespace, and valid binomial structure.

Examples

```
validate_species_names(c(  
  "Homo sapiens",  
  "Chilina sp.",  
  NA,  
  ""  
)
```

Index

`assess_taxonomic_match`, [2](#)
`assess_taxonomic_match()`, [8](#), [16](#)

`classify_scientific_names`, [3](#)
`clean_scientific_names`, [5](#)

`export_taxonomic_results`, [6](#)

`flag_taxonomic_review`, [7](#)
`flag_taxonomic_review()`, [16](#)

`parse_scientific_names`, [9](#)
`prepare_taxonomic_queries`, [10](#)

`resolve_taxonomy`, [11](#)
`resolve_taxonomy()`, [2](#), [16](#)

`standardize_scientific_names`, [12](#)
`summarize_taxonomic_resolution`, [13](#)
`summarize_taxonomic_resolution()`, [15](#)

`taxonomic_resolution_report`, [15](#)
`taxresolve`, [16](#)
`taxresolve()`, [6](#), [15](#)

`validate_species_names`, [17](#)