

Package ‘overturemapsr’

July 22, 2025

Title Download Overture Maps Data in R

Version 0.1.0

Description Overture Maps offers free and open geospatial map data sourced from various providers and standardized to a common schema. This tool allows you to download Overture Maps data for a specific region of interest and convert it to several different file formats. For more information, visit <<https://overturemaps.org/download/>>.

License MIT + file LICENSE

URL <https://github.com/denironyx/overturemapsr>

BugReports <https://github.com/denironyx/overturemapsr/issues>

Encoding UTF-8

RoxygenNote 7.3.1

Depends R (>= 3.5.0)

Imports arrow, dplyr, sf

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Config/Needs/website rmarkdown

NeedsCompilation no

Author Dennis Irerere [aut, cre, cph]

Maintainer Dennis Irerere <denironyx@gmail.com>

Repository CRAN

Date/Publication 2025-02-14 16:20:11 UTC

Contents

dataset_path	2
get_all_overture_schema_types	2
get_dataset_metadata	3
record_batch_reader	4

Index	5
--------------	----------

dataset_path	<i>dataset_path</i>
--------------	---------------------

Description

This function returns the S3 path for the specified Overture dataset schema type.

Usage

```
dataset_path(schema_type, release_date = "2025-01-22")
```

Arguments

- schema_type Character. Required. The type of feature to select. Examples include 'building', 'place', etc. To learn more, run get_all_overture_schema_types().
- release_date Character. Optional. The dataset release date (format: 'YYYY-MM-DD'). Defaults to the latest available release.

Value

Character. The S3 path to the bucket where the data is stored.

Examples

```
# Example usage
path <- dataset_path('place', release_date = '2025-01-22')
print(path)
```

get_all_overture_schema_types
<i>get_all_overture_types</i>

Description

This function returns all of the overturemaps theme types.

Usage

```
get_all_overture_schema_types()
```

Value

Character vector. All overturemaps theme types.

Note

The theme types are important for fetching data from the S3 bucket, as they indicate if you are fetching places, buildings, admin, etc.

Examples

```
# Example usage
types <- get_all_overture_schema_types()
print(types)
```

get_dataset_metadata	<i>get_dataset_metadata</i>
----------------------	-----------------------------

Description

This function retrieves metadata information for a given dataset type.

Usage

```
get_dataset_metadata(schema_type, release_date = "2025-01-22")
```

Arguments

schema_type	Character. Required. The type of feature to select.
release_date	Character. Optional. The dataset release date (format: 'YYYY-MM-DD').

Value

A list containing metadata such as column names and dataset size.

Examples

```
metadata <- get_dataset_metadata('place', release_date = '2025-01-22')
print(metadata)
```

record_batch_reader	<i>record_batch_reader</i>
---------------------	----------------------------

Description

This function retrieves a filtered dataset from the specified Overture dataset type, optionally within a bounding box, and converts it to an `sf` object.

Usage

```
record_batch_reader(schema_type, bbox = NULL, release_date = "2025-01-22")
```

Arguments

<code>schema_type</code>	Character. Required. The type of feature to select. Examples include 'building', 'place', etc. To learn more, run <code>get_all_overture_schema_types()</code> .
<code>bbox</code>	Numeric vector. Optional. A bounding box specified as <code>c(xmin, ymin, xmax, ymax)</code> . It is recommended to use a bounding box to limit the dataset size and processing time. Without a bounding box, processing the entire dataset (e.g., buildings over 2 billion) can be time-consuming.
<code>release_date</code>	Character. Optional. The dataset release date (format: 'YYYY-MM-DD'). Defaults to the latest available release.

Value

An `sf` object containing the filtered dataset based on the bounding box.

Examples

```
# Example usage with a bounding box takes > 20 secs
sf_bbox <- c(-122.5, 37.7, -122.3, 37.8)
result <- record_batch_reader(schema_type = 'place', bbox = sf_bbox)
print(result)
```

Index

`dataset_path`, [2](#)

`get_all_overture_schema_types`, [2](#)

`get_dataset_metadata`, [3](#)

`record_batch_reader`, [4](#)