

# Package ‘hfhub’

April 15, 2026

**Title** Hugging Face Hub Interface

**Version** 0.1.2

**Description** Provides functionality to download and cache files from 'Hugging Face Hub' <<https://huggingface.co/models>>.

Uses the same caching structure so files can be shared between different client libraries.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** httr, filelock, fs, cli, withr, curl, glue, urltools, rlang

**Suggests** testthat (>= 3.0.0), jsonlite

**Config/testthat/edition** 3

**URL** <https://mlverse.github.io/hfhub/>

**BugReports** <https://github.com/mlverse/hfhub/issues>

**NeedsCompilation** no

**Author** Tomasz Kalinowski [ctb, cre],

Daniel Falbel [aut],

Christophe Regouby [ctb],

Posit Software, PBC [cph, fnd] (ROR: <<https://ror.org/03wc8by49>>)

**Maintainer** Tomasz Kalinowski <[tomasz@posit.co](mailto:tomasz@posit.co)>

**Repository** CRAN

**Date/Publication** 2026-04-15 14:50:02 UTC

## Contents

hub_download . . . . .	2
hub_repo_info . . . . .	3
hub_snapshot . . . . .	4
WEIGHTS_NAME . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

---

hub_download	<i>Downloads files from HuggingFace repositories</i>
--------------	--

---

## Description

Downloads files from HuggingFace repositories

## Usage

```
hub_download(  
  repo_id,  
  filename,  
  ...,  
  revision = "main",  
  repo_type = "model",  
  local_files_only = FALSE,  
  force_download = FALSE  
)
```

## Arguments

repo_id	The repository identifier, eg "bert-base-uncased" or "deepset/sentence_bert".
filename	Filename to download from the repository. Example "config.json".
...	currently unused.
revision	Revision (branch, tag or commitid) to download the file from.
repo_type	The type of the repository. Currently only "model" is supported.
local_files_only	Only use cached files?
force_download	For re-downloading of files that are cached.

## Value

The file path of the downloaded or cached file. The snapshot path is returned as an attribute.

## Examples

```
try({  
  withr::with_envvar(c(HUGGINGFACE_HUB_CACHE = tempdir()), {  
    path <- hub_download("gpt2", "config.json")  
    print(path)  
    str(jsonlite::fromJSON(path))  
  })  
})
```

---

`hub_repo_info`*Queries information about Hub repositories*

---

**Description**

Queries information about Hub repositories

**Usage**

```
hub_repo_info(  
  repo_id,  
  ...,  
  repo_type = NULL,  
  revision = NULL,  
  files_metadata = FALSE  
)
```

```
hub_dataset_info(repo_id, ..., revision = NULL, files_metadata = FALSE)
```

**Arguments**

<code>repo_id</code>	The repository identifier, eg "bert-base-uncased" or "deepset/sentence_bert".
<code>...</code>	currently unused.
<code>repo_type</code>	The type of the repository. Currently only "model" is supported.
<code>revision</code>	Revision (branch, tag or commitid) to download the file from.
<code>files_metadata</code>	Obtain files metadata information when querying repository information.

**Value**

A list with information about the repository, including model details, file siblings, tags, and other metadata returned by the Hugging Face API.

**Functions**

- `hub_dataset_info()`: Query information from a Hub Dataset

**Examples**

```
try({  
  info <- hub_repo_info("gpt2")  
  info$modelId  
})
```

---

hub_snapshot	<i>Snapshot the entire repository</i>
--------------	---------------------------------------

---

### Description

Downloads and stores all files from a Hugging Face Hub repository.

### Usage

```
hub_snapshot(
  repo_id,
  ...,
  revision = "main",
  repo_type = "model",
  local_files_only = FALSE,
  force_download = FALSE,
  allow_patterns = NULL,
  ignore_patterns = NULL
)
```

### Arguments

repo_id	The repository identifier, eg "bert-base-uncased" or "deepset/sentence_bert".
...	currently unused.
revision	Revision (branch, tag or commitid) to download the file from.
repo_type	The type of the repository. Currently only "model" is supported.
local_files_only	Only use cached files?
force_download	For re-downloading of files that are cached.
allow_patterns	A character vector containing patters that are used to filter allowed files to snapshot.
ignore_patterns	A character vector containning patterns to reject files from being downloaded.

### Value

A string with the path to the snapshot directory containing all downloaded files.

### Examples

```
try({
  withr::with_envvar(c(HUGGINGFACE_HUB_CACHE = tempdir()), {
    path <- hub_snapshot("gpt2", allow_patterns = "config.json")
    list.files(path)
  })
})
```

---

WEIGHTS_NAME	<i>Weight file names in HUB</i>
--------------	---------------------------------

---

**Description**

Weight file names in HUB

**Usage**

WEIGHTS\_NAME()

WEIGHTS\_INDEX\_NAME()

**Value**

A string with the default file names for indexes in the Hugging Face Hub.

**Functions**

- WEIGHTS\_NAME(): Name of weights file
- WEIGHTS\_INDEX\_NAME(): Name of weights index file

**Examples**

WEIGHTS\_NAME()

WEIGHTS\_INDEX\_NAME()

# Index

hub\_dataset\_info (hub\_repo\_info), [3](#)  
hub\_download, [2](#)  
hub\_repo\_info, [3](#)  
hub\_snapshot, [4](#)

WEIGHTS\_INDEX\_NAME (WEIGHTS\_NAME), [5](#)  
WEIGHTS\_NAME, [5](#)