

Package: safetensors (via r-universe)

October 17, 2024

Title Safetensors File Format

Version 0.1.2.9000

Description A file format for storing tensors that is secure (doesn't allow for code execution), fast and simple to implement. 'safetensors' also enables cross language and cross frameworks compatibility making it an ideal format for storing machine learning model weights.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests testthat (>= 3.0.0), torch (>= 0.11.0)

Config/testthat/edition 3

Imports cli, jsonlite, R6, rlang

URL <https://github.com/mlverse/safetensors>,
<https://mlverse.github.io/safetensors/>

BugReports <https://github.com/mlverse/safetensors/issues>

Repository <https://mlverse.r-universe.dev>

RemoteUrl <https://github.com/mlverse/safetensors>

RemoteRef HEAD

RemoteSha 00d64fc0040ee80ac37ffdaaa58076fa343990e0

Contents

safetensors	2
safe_load_file	3
safe_save_file	4

Index	6
--------------	----------

safetensors

Low level control over safetensors files

Description

Low level control over safetensors files

Low level control over safetensors files

Details

Allows opening a connection to a safetensors file and query the tensor names, metadata, etc. Opening a connection only reads the file metadata into memory. This allows for more fined grained control over reading.

Public fields

`con` the connection object with the file

`metadata` an R list containing the metadata header in the file

`framework` the framework used to return the tensors

`device` the device to where tensors are copied

`max_offset` the largest offset boundary that was visited. Mainly used in torch to find the end of the safetensors file.

Methods

Public methods:

- [safetensors\\$new\(\)](#)
- [safetensors\\$keys\(\)](#)
- [safetensors\\$get_tensor\(\)](#)
- [safetensors\\$clone\(\)](#)

Method `new()`: Opens the connection with the file

Usage:

```
safetensors$new(path, ..., framework = "torch", device = "cpu")
```

Arguments:

`path` Path to the file to load

`...` Unused

`framework` Framework to load the data into. Currently only torch is supported

`device` Device to copy data once loaded

Method `keys()`: Get the keys (tensor names) in the file

Usage:

```
safetensors$keys()
```

Method `get_tensor()`: Get a tensor from its name

Usage:

```
safetensors$get_tensor(name)
```

Arguments:

name Name of the tensor to load

Method `clone()`: The objects of this class are cloneable with this method.

Usage:

```
safetensors$clone(deep = FALSE)
```

Arguments:

deep Whether to make a deep clone.

Examples

```
if (rlang::is_installed("torch") && torch::torch_is_installed()) {
  tensors <- list(x = torch::torch_randn(10, 10))
  temp <- tempfile()
  safe_save_file(tensors, temp)
  f <- safetensors$new(temp)
  f$get_tensor("x")
}
```

safe_load_file

Safe load a safetensors file

Description

Loads an safetensors file from disk.

Usage

```
safe_load_file(path, ..., framework = "torch", device = "cpu")
```

Arguments

path	Path to the file to load
...	Unused
framework	Framework to load the data into. Currently only torch is supported
device	Device to copy data once loaded

Value

A list with tensors in the file. The metadata attribute can be used to find metadata the metadata header in the file.

See Also

[safetensors](#), [safe_save_file\(\)](#)

Examples

```
if (rlang::is_installed("torch") && torch::torch_is_installed()) {
  tensors <- list(x = torch::torch_randn(10, 10))
  temp <- tempfile()
  safe_save_file(tensors, temp)
  safe_load_file(temp)
}
```

safe_save_file	<i>Writes a list of tensors to the safetensors format</i>
----------------	---

Description

Writes a list of tensors to the safetensors format

Usage

```
safe_save_file(tensors, path, ..., metadata = NULL)
```

```
safe_serialize(tensors, ..., metadata = NULL)
```

Arguments

tensors	A named list of tensors. Currently only torch tensors are supported.
path	The path to save the tensors to. It can also be a binary connection, as eg. created with <code>file()</code> .
...	Currently unused.
metadata	An optional string that is added to the file header. Possibly adding additional description to the weights.

Value

The path invisibly or a raw vector.

Functions

- `safe_serialize()`: Serializes the tensors and returns a raw vector.

Examples

```
if (rlang::is_installed("torch") && torch::torch_is_installed()) {  
  tensors <- list(x = torch::torch_randn(10, 10))  
  temp <- tempfile()  
  safe_save_file(tensors, temp)  
  safe_load_file(temp)  
  
  ser <- safe_serialize(tensors)  
}
```

Index

`safe_load_file`, [3](#)
`safe_save_file`, [4](#)
`safe_save_file()`, [4](#)
`safe_serialize(safe_save_file)`, [4](#)
`safetensors`, [2](#), [4](#)