
brotlipy Documentation

Release 0.7.0

Cory Benfield

Jun 12, 2017

Contents

1	Documentation	3
1.1	Installation	3
1.2	API Documentation	3
2	License	7
	Python Module Index	9

Brotlipy is a collection of CFFI-based bindings to the Brotli compression reference implementation as written by Google. This enables Python software to easily and quickly work with the Brotli compression algorithm, regardless of what interpreter is being used.

Brotlipy has a very similar interface to the standard library's `zlib` module:

```
import brotli

# Decompress a Brotli-compressed payload in one go.
decompressed_data = brotli.decompress(compressed_data)

# Alternatively, you can do incremental decompression.
d = brotli.Decompressor()
for chunk in chunks_of_compressed_data:
    some_uncompressed_data = d.decompress(chunk)

remaining_data = d.flush()

# You can compress data too.
compressed = brotli.compress(uncompressed_data)
```

For more details on the API, see [API Documentation](#).

Installation

Installing Brotlipy couldn't be easier:

```
$ pip install brotlipy
```

On OS X and Windows this should succeed without difficulty. On Linux, the above command has a few dependencies: mostly, you need a C compiler, the Python header files, and libffi.

On Debian-based systems, you can obtain these files by running:

```
$ apt-get install build-essential python-dev libffi-dev
```

On Red Hat-based systems, you can obtain these files by running:

```
$ yum install gcc libffi-devel python-devel
```

API Documentation

This section of the documentation covers the API of brotlipy.

Decompression

`brotli.decompress` (*data*)

Decompress a complete Brotli-compressed string.

Parameters `data` – A bytestring containing Brotli-compressed data.

`class brotli.Decompressor` (*dictionary*=''')

An object that allows for streaming decompression of Brotli-compressed data.

Changed in version 0.5.0: Added `dictionary` parameter.

Parameters `dictionary` (bytes) – A pre-set dictionary for LZ77. Please use this with caution: if a dictionary is used for compression, the same dictionary **must** be used for decompression!

decompress (*data*)

Decompress part of a complete Brotli-compressed string.

Parameters `data` – A bytestring containing Brotli-compressed data.

Returns A bytestring containing the decompressed data.

finish ()

Finish the decompressor. As the decompressor decompresses eagerly, this will never actually emit any data. However, it will potentially throw errors if a truncated or damaged data stream has been used.

Note that, once this method is called, the decompressor is no longer safe for further use and must be thrown away.

flush ()

Complete the decompression, return whatever data is remaining to be decompressed.

Deprecated since version 0.4.0: This method is no longer required, as `decompress()` will now decompress eagerly.

Returns A bytestring containing the remaining decompressed data.

Compression

`brotli.compress` (*data*, *mode*=<*BrotliEncoderMode.GENERIC*: 0>, *quality*=11, *lgwin*=22, *lgblock*=0, *dictionary*='')

Compress a string using Brotli.

Changed in version 0.5.0: Added `mode`, `quality`, `lgwin`, `lgblock`, and `dictionary` parameters.

Parameters

- **data** (bytes) – A bytestring containing the data to compress.
- **mode** (*BrotliEncoderMode* or int) – The encoder mode.
- **quality** (int) – Controls the compression-speed vs compression-density tradeoffs. The higher the quality, the slower the compression. The range of this value is 0 to 11.
- **lgwin** (int) – The base-2 logarithm of the sliding window size. The range of this value is 10 to 24.
- **lgblock** (int) – The base-2 logarithm of the maximum input block size. The range of this value is 16 to 24. If set to 0, the value will be set based on `quality`.
- **dictionary** (bytes) – A pre-set dictionary for LZ77. Please use this with caution: if a dictionary is used for compression, the same dictionary **must** be used for decompression!

Returns The compressed bytestring.

Return type bytes

`class brotli.Compressor` (*mode*=<*BrotliEncoderMode.GENERIC*: 0>, *quality*=11, *lgwin*=22, *lgblock*=0, *dictionary*='')

An object that allows for streaming compression of data using the Brotli compression algorithm.

New in version 0.5.0.

Parameters

- **mode** (*BrotliEncoderMode* or *int*) – The encoder mode.
- **quality** (*int*) – Controls the compression-speed vs compression-density tradeoffs. The higher the quality, the slower the compression. The range of this value is 0 to 11.
- **lgwin** (*int*) – The base-2 logarithm of the sliding window size. The range of this value is 10 to 24.
- **lgblock** (*int*) – The base-2 logarithm of the maximum input block size. The range of this value is 16 to 24. If set to 0, the value will be set based on *quality*.
- **dictionary** (*bytes*) – A pre-set dictionary for LZ77. Please use this with caution: if a dictionary is used for compression, the same dictionary **must** be used for decompression!

compress (*data*)

Incrementally compress more data.

Parameters *data* – A bytestring containing data to compress.

Returns A bytestring containing some compressed data. May return the empty bytestring if not enough data has been inserted into the compressor to create the output yet.

finish ()

Finish the compressor. This will emit the remaining output data and transition the compressor to a completed state. The compressor cannot be used again after this point, and must be replaced.

flush ()

Flush the compressor. This will emit the remaining output data, but will not destroy the compressor. It can be used, for example, to ensure that given chunks of content will decompress immediately.

class `brotli.BrotliEncoderMode`

Compression modes for the Brotli encoder.

New in version 0.5.0.

FONT = 2

Compression mode used in WOFF 2.0

GENERIC = 0

Default compression mode. The compressor does not know anything in advance about the properties of the input.

TEXT = 1

Compression mode for UTF-8 format text input.

Errors

class `brotli.Error`

Raised whenever an error is encountered with compressing or decompressing data using brotli.

New in version 0.5.1.

`brotli.error = <class 'brotli.brotli.Error'>`

An alias of *Error* that exists for compatibility with the original C brotli module.

CHAPTER 2

License

Brotlipy's source code is made available under the MIT license. Brotli itself is licensed under Version 2.0 of the Apache Software License.

b

brotli, 3

B

brotili (module), 3
BrotliEncoderMode (class in brotili), 5

C

compress() (brotili.brotli method), 4
compress() (brotili.Compressor method), 5
Compressor (class in brotili), 4

D

decompress() (brotili.brotli method), 3
decompress() (brotili.Decompressor method), 4
Decompressor (class in brotili), 3

E

Error (class in brotili), 5
error (in module brotili), 5

F

finish() (brotili.Compressor method), 5
finish() (brotili.Decompressor method), 4
flush() (brotili.Compressor method), 5
flush() (brotili.Decompressor method), 4
FONT (brotili.BrotliEncoderMode attribute), 5

G

GENERIC (brotili.BrotliEncoderMode attribute), 5

T

TEXT (brotili.BrotliEncoderMode attribute), 5