textunwrap Documentation

Release 0.1.3

Julien Palard

Mar 03, 2021

Contents

1 textunwrap **3**
1.1 Credits . . . 3

2 Installation **5**
2.1 Stable release . . . 5
2.2 From sources . . . 5

3 Usage **7**
3.1 Command line tools . . . 7

4 Contributing **9**
4.1 Types of Contributions . . . 9
4.2 Get Started! . . . 9
4.3 Pull Request Guidelines . . . 10

5 Credits **11**
5.1 Development Lead . . . 11
5.2 Contributors . . . 11

6 Indices and tables **13**

Contents:

CHAPTER 1

textunwrap

Tries to reverse `textwrap.wrap()`.

- Documentation: https://textunwrap.readthedocs.io.
- Free software: MIT

1.1 Credits

This package was created with Cookiecutter and the audreyr/cookiecutter-pypackage project template.

CHAPTER 2

Installation

2.1 Stable release

To install textunwrap, run this command in your terminal:

```
$ python3 -m pip install textunwrap
```

This is the preferred method to install textunwrap, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

2.2 From sources

The sources for textunwrap can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/JulienPalard/textunwrap
```

Or download the tarball:

```
$ curl  -OL https://github.com/JulienPalard/textunwrap/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python3 setup.py install
```

CHAPTER 3

Usage

To use textunwrap in a project, use the `unwrap` function exposed by the `textunwrap` package:

```
>>> from textwrap import fill
>>> from textunwrap import unwrap
>>> print(fill("foo " * 100))
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
>>> print(unwrap(fill("foo " * 100)))
foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo␣
↪foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo␣
↪foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo␣
↪foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo␣
↪foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo foo
```

3.1 Command line tools

A `textunwrap` (and a `textwrap`) command line are provided:

```
$ cat APAFML-wrapped.txt
Copyright (c) 1985, 1987, 1989, 1990, 1991, 1992, 1993, 1997 Adobe Systems
Incorporated. All Rights Reserved.

This file and the 14 PostScript(R) AFM files it accompanies may be used,
copied, and distributed for any purpose and without charge, with or without
modification, provided that all copyright notices are retained; that the AFM
files are not distributed without this file; that all modifications to this
file or any of the AFM files are prominently noted in the modified file(s); and
that this paragraph is not modified. Adobe Systems has no responsibility or
```

(continues on next page)

(continued from previous page)

```
obligation to support the use of the AFM files.

$ textunwrap APAFML-wrapped.txt
Copyright (c) 1985, 1987, 1989, 1990, 1991, 1992, 1993, 1997 Adobe Systems␣
↪Incorporated. All Rights Reserved.

This file and the 14 PostScript(R) AFM files it accompanies may be used, copied, and␣
↪distributed for any purpose and without charge, with or without modification,␣
↪provided that all copyright notices are retained; that the AFM files are not␣
↪distributed without this file; that all modifications to this file or any of the␣
↪AFM files are prominently noted in the modified file(s); and that this paragraph is␣
↪not modified. Adobe Systems has no responsibility or obligation to support the use␣
↪of the AFM files.
```

CHAPTER 4

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/JulienPalard/textunwrap/issues.

If you are reporting a bug, please include detailed steps to reproduce the bug, if possible the file used, or better a minimum sample file reproducing the bug.

4.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" and "help wanted" is open to whoever wants to implement it.

4.1.3 Enhance the algorithm

The current unwrapping algorithm is pretty dumb, any pull requests to enhance it are welcome.

4.2 Get Started!

Ready to contribute? Here's how to set up *textunwrap* for local development.

1. Fork the *textunwrap* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/textunwrap.git
```

3. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

4. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 textunwrap tests
$ pytest
```

6. Commit your changes and push your branch to GitHub:

```
$ git add -p
$ git commit -m "Your detailed description of your changes."
$ git push -u origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/JulienPalard/textunwrap/pull_requests and make sure that the tests pass for all supported Python versions.

CHAPTER 5

Credits

5.1 Development Lead

- Julien Palard <julien@palard.fr>

5.2 Contributors

None yet. Why not be the first?

CHAPTER 6

Indices and tables

- genindex
- modindex
- search