

---

# Wordle Documentation

*Release 0.1.4*

**Hrishikesh Terdalkar**

**May 17, 2022**



**CONTENTS:**

<b>1</b>	<b>Features</b>	<b>3</b>
<b>2</b>	<b>Example</b>	<b>5</b>
2.1	Helper Session . . . . .	5
<b>3</b>	<b>Terminology</b>	<b>7</b>
<b>4</b>	<b>Solver</b>	<b>9</b>
<b>5</b>	<b>Helper</b>	<b>11</b>
<b>6</b>	<b>Evaluation</b>	<b>13</b>
6.1	Setup . . . . .	13
6.2	Performance . . . . .	13
<b>7</b>	<b>Indices and tables</b>	<b>23</b>
	<b>Python Module Index</b>	<b>25</b>
	<b>Index</b>	<b>27</b>



Wordle – Game, Solver and Helper!

- Free software: GNU General Public License v3
- Documentation: <https://python-wordle.readthedocs.io>.



## FEATURES

- **Comprehensive CLI Application**
  - Play Wordle in your terminal!
  - Let the solver play it out for you!
  - Use the helper mode for solving Wordles provided by other platforms!
- **New Wordle every day.**
  - **Disclaimer:** This has nothing to do with the NYTimes Wordle.
- Play more using `--random` option.
- Reproducible random Wordle by specifying `--seed`.
- `WordleSolver()` class for solving Wordle.
- `--solve` option to let `WordleSolver()` do its work in front of your eyes!
- `--helper` option to activate the helper mode for help with solving Wordles provided by other platforms.





## EXAMPLE



### 2.1 Helper Session

```
Attempt 1
=====

Suggestions: [('kioea', 95.7343887423043)]
Chosen word: aeons
Obtained result: 01201
Eliminated 15868 options.

Attempt 2
=====

Suggestions: [('thrip', 78.0), ('thirl', 78.0)]
Chosen word: thrip
Obtained result: 00000
Eliminated 39 options.

Attempt 3
```

(continues on next page)

(continued from previous page)

```
=====
```

```
Suggestions: [('blayk', 72.72727272727273), ('milky', 72.72727272727273), ('bulky', 72.  
↪72727272727273), ('lucky', 72.72727272727273), ('mckay', 72.72727272727273)]
```

```
Chosen word: milky
```

```
Obtained result: 10020
```

```
Eliminated 10 options.
```

```
Attempt 4
```

```
=====
```

```
Valid Words: {'smoke'}
```

```
Suggestions: [('smoke', 0.0)]
```

```
Chosen word: smoke
```

```
Obtained result: 22222
```

```
Congratulations!
```

## TERMINOLOGY

All of the following terms are valid in the context of a specific Wordle.

- A set of **known letters** is a set of letters which are known to be present in the Wordle's word based on clues obtained from the previous attempts.
- A set of **valid words** at any given point refers to a set of words from vocabulary that are possible solutions to the Wordle at that point based on the clues obtained till then.
- A letter is said to **cover** a word if the letter belongs to that word.
- **Coverage** (*absolute*) of a word refers to the percentage of valid words (at that point) covered by the letters in that word.
- **Coverage** (*relative*) of a word refers to the percentage of valid words (at that point) covered by the set of



## SOLVER

Solver roughly uses the following strategy,

- Guess a word with highest (relative) coverage.
- **After each guess, eliminate words based on the clues, e.g.,**
  - Remove all words that contain an `incorrect-letter`
  - Remove all words that do not contain the identified (`correct-letter`, `correct-position`) combinations
  - Remove all words that contain the identified (`correct-letter`, `incorrect-position`) combinations
  - Remove all words that do not contain a `correct-letter`
  - Re-calculate (relative) coverage and guess again
- If at any point, number of valid words drops below number of attempts left, guess the valid words one-by-one.
- Additionally, there are heuristics to choose a word if multiple words with best coverage-score exist.



## HELPER

Helper mode can be used to get assistance while solving Wordle from another provider.

- Suggestions for words will be provided at each step.
- You may choose a word and obtain result.
- **Enter the obtained result as a ternary string, i.e., a string made of 0, 1, and 2.**
  - 0 : (grey) : an incorrect letter.
  - 1 : (yellow) : a correct letter in wrong position.
  - 2 : (green) : a correct letter in correct position.
- e.g., if the third letter turned green and fifth letter turned yellow in the third party Wordle, then you should input 00201 as the result obtained.





## EVALUATION

WordleSolver has been evaluated on the entire vocabulary.

### 6.1 Setup

- Word Size: 5
- Vocabulary Size: 15918
- Maximum Attempts: 6

### 6.2 Performance

- Success Count: 15446
- Success Rate: 97.03 %
- Average Attempts: 4.58

#### 6.2.1 Wordle

Wordle – Game, Solver and Helper!

- Free software: GNU General Public License v3
- Documentation: <https://python-wordle.readthedocs.io>.

## Features

- **Comprehensive CLI Application**
  - Play Wordle in your terminal!
  - Let the solver play it out for you!
  - Use the helper mode for solving Wordles provided by other platforms!
- **New Wordle every day.**
  - **Disclaimer:** This has nothing to do with the NYTimes Wordle.
- Play more using `--random` option.
- Reproducible random Wordle by specifying `--seed`.
- `WordleSolver()` class for solving Wordle.
- `--solve` option to let `WordleSolver()` do its work in front of your eyes!
- `--helper` option to activate the helper mode for help with solving Wordles provided by other platforms.

## Install

To install, run this command in your terminal:

```
$ pip install python-wordle
```

## Usage

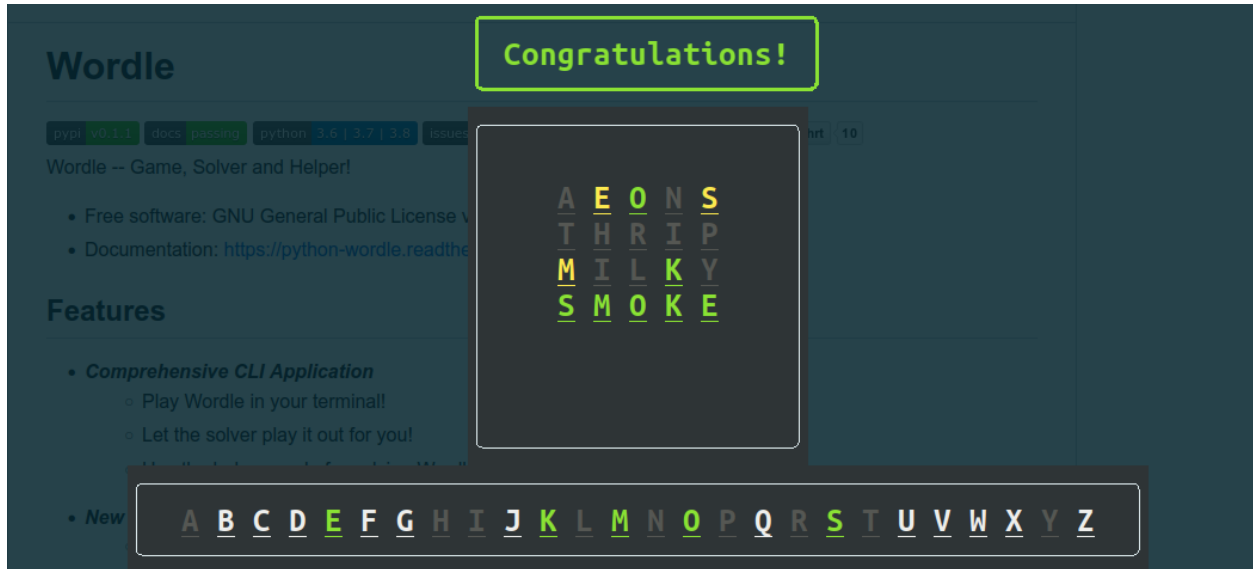
```
usage: wordle [-h] [--random] [--seed SEED] [--solve] [--helper]
```

Wordle on your terminal

optional arguments:

```
-h, --help    show this help message and exit
--random      Show a random Wordle
--seed SEED   Seed the RNG
--solve       Simulate WordleSolver
--helper      Take help from WordleSolver for a Wordle in a different platform
```

## Example



## Helper Session

Attempt 1

=====

Suggestions: [('kioea', 95.7343887423043)]

Chosen word: aeons

Obtained result: 01201

Eliminated 15868 options.

Attempt 2

=====

Suggestions: [('thrip', 78.0), ('thirl', 78.0)]

Chosen word: thrip

Obtained result: 00000

Eliminated 39 options.

Attempt 3

=====

Suggestions: [('blayk', 72.72727272727273), ('milky', 72.72727272727273), ('bulky', 72.72727272727273), ('lucky', 72.72727272727273), ('mckay', 72.72727272727273)]

Chosen word: milky

Obtained result: 10020

Eliminated 10 options.

Attempt 4

=====

Valid Words: {'smoke'}

(continues on next page)

(continued from previous page)

```
Suggestions: [('smoke', 0.0)]  
Chosen word: smoke  
Obtained result: 22222  
  
Congratulations!
```

### Terminology

All of the following terms are valid in the context of a specific Wordle.

- A set of **known letters** is a set of letters which are known to be present in the Wordle's word based on clues obtained from the previous attempts.
- A set of **valid words** at any given point refers to a set of words from vocabulary that are possible solutions to the Wordle at that point based on the clues obtained till then.
- A letter is said to **cover** a word if the letter belongs to that word.
- **Coverage** (*absolute*) of a word refers to the percentage of valid words (at that point) covered by the letters in that word.
- **Coverage** (*relative*) of a word refers to the percentage of valid words (at that point) covered by the set of

### Solver

Solver roughly uses the following strategy,

- Guess a word with highest (relative) coverage.
- **After each guess, eliminate words based on the clues, e.g.,**
  - Remove all words that contain an **incorrect-letter**
  - Remove all words that do not contain the identified (**correct-letter**, **correct-position**) combinations
  - Remove all words that contain the identified (**correct-letter**, **incorrect-position**) combinations
  - Remove all words that do not contain a **correct-letter**
  - Re-calculate (relative) coverage and guess again
- If at any point, number of valid words drops below number of attempts left, guess the valid words one-by-one.
- Additionally, there are heuristics to choose a word if multiple words with best coverage-score exist.

### Helper

Helper mode can be used to get assistance while solving Wordle from another provider.

- Suggestions for words will be provided at each step.
- You may choose a word and obtain result.
- **Enter the obtained result as a ternary string, i.e., a string made of 0, 1, and 2.**
  - 0 : (grey) : an incorrect letter.
  - 1 : (yellow) : a correct letter in wrong position.

- 2 : (green) : a correct letter in correct position.
- e.g., if the third letter turned green and fifth letter turned yellow in the third party Wordle, then you should input 00201 as the result obtained.

## Evaluation

WordleSolver has been evaluated on the entire vocabulary.

## Setup

- Word Size: 5
- Vocabulary Size: 15918
- Maximum Attempts: 6

## Performance

- Success Count: 15446
- Success Rate: 97.03 %
- Average Attempts: 4.58

## 6.2.2 Installation

### Stable release

To install Wordle, run this command in your terminal:

```
$ pip install python_wordle
```

This is the preferred method to install Wordle, 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.

### From sources

The sources for Wordle can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/hrishikeshrt/python-wordle
```

Or download the [tarball](#):

```
$ curl -OJL https://github.com/hrishikeshrt/python-wordle/tarball/master
```

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

```
$ python setup.py install
```

### 6.2.3 Usage

```
usage: wordle [-h] [--random] [--seed SEED] [--solve] [--helper]
```

Wordle on your terminal

optional arguments:

```
-h, --help    show this help message and exit
--random      Show a random Wordle
--seed SEED   Seed the RNG
--solve       Simulate WordleSolver
--helper      Take help from WordleSolver for a Wordle in a different platform
```

### 6.2.4 wordle

**wordle package**

**Submodules**

**wordle.cli module**

**wordle.defaults module**

Wordle Defaults

**wordle.evaluate module**

**wordle.solver module**

**wordle.vocab module**

Wordle Vocabulary

```
class wordle.vocab.Vocabulary(alphabet: Iterable = 'abcdefghijklmnopqrstuvwxyz', words_file: str =
                               PosixPath('/home/docs/checkouts/readthedocs.org/user_builds/python-
                               wordle/checkouts/latest/wordle/data/vocabulary.txt'), word_length: int = 5,
                               vocab_cache: str =
                               PosixPath('/home/docs/checkouts/readthedocs.org/user_builds/python-
                               wordle/checkouts/latest/wordle/data/vocabulary.json'), index_cache: str =
                               PosixPath('/home/docs/checkouts/readthedocs.org/user_builds/python-
                               wordle/checkouts/latest/wordle/data/index.json'))
```

Bases: object

**build\_vocabulary**(*use\_cache=True*)

**build\_index**(*use\_cache=True*)

**is\_word**(*text*)

## wordle.wordle module

Wordle for Python

```
class wordle.wordle.Wordle(word: Optional[str] = None, max_attempts: int = 6, seed: int = 'today', display:
                             bool = False)
```

Bases: object

**message**(msg, style='bold yellow')

**guess**(word)

**show**(style=None)

## Module contents

Top-level package for python-wordle

## 6.2.5 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:

### Types of Contributions

#### Report Bugs

Report bugs at <https://github.com/hrishikeshrt/python-wordle/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

#### Fix Bugs

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

#### Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

## Write Documentation

Wordle could always use more documentation, whether as part of the official Wordle docs, in docstrings, or even on the web in blog posts, articles, and such.

## Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/hrishikeshrt/python-wordle/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## Get Started!

Ready to contribute? Here's how to set up *python-wordle* for local development.

1. Fork the *python-wordle* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/python-wordle.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv python_wordle
$ cd python_wordle/
$ python setup.py develop
```

4. Create a branch for local development:

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

Now you can make your changes locally.

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

```
$ flake8 python_wordle tests
$ python setup.py test or pytest
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

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

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

7. Submit a pull request through the GitHub website.



## 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 3.5, 3.6, 3.7 and 3.8, and for PyPy. Check [https://travis-ci.com/hrishikeshrt/python\\_wordle/pull\\_requests](https://travis-ci.com/hrishikeshrt/python_wordle/pull_requests) and make sure that the tests pass for all supported Python versions.

## Tips

To run a subset of tests:

```
$ pytest tests.test_wordle
```

## Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bump2version patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.

## 6.2.6 Credits

### Development Lead

- Hrishikesh Terdalkar <[hrishikeshrt@linuxmail.org](mailto:hrishikeshrt@linuxmail.org)>

### Contributors

None yet. Why not be the first?

## 6.2.7 History

### 0.1.0 (2022-02-28)

- First release on PyPI.



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`



## PYTHON MODULE INDEX

### W

`wordle`, [19](#)  
`wordle.defaults`, [18](#)  
`wordle.vocab`, [18](#)  
`wordle.wordle`, [19](#)



## INDEX

### B

`build_index()` (*wordle.vocab.Vocabulary method*), 18  
`build_vocabulary()` (*wordle.vocab.Vocabulary method*), 18

### G

`guess()` (*wordle.wordle.Wordle method*), 19

### I

`is_word()` (*wordle.vocab.Vocabulary method*), 18

### M

`message()` (*wordle.wordle.Wordle method*), 19

module

- `wordle`, 19
- `wordle.defaults`, 18
- `wordle.vocab`, 18
- `wordle.wordle`, 19

### S

`show()` (*wordle.wordle.Wordle method*), 19

### V

`Vocabulary` (*class in wordle.vocab*), 18

### W

wordle

- module, 19

`Wordle` (*class in wordle.wordle*), 19

`wordle.defaults`

- module, 18

`wordle.vocab`

- module, 18

`wordle.wordle`

- module, 19