# jwalk Documentation

*Release 0.5.1*

**Kamil Sindi**

**Apr 23, 2017**

# Contents

**1 jwalk** **3**
1.1 Install . . . . . 3
1.2 Build . . . . . 3
1.3 Usage . . . . . 3
1.4 Test . . . . . 4
1.5 Blog . . . . . 4
1.6 License . . . . . 4
1.7 References . . . . . 4

**2 Changelog** **5**
2.1 Releases . . . . . 5

**3 Indices and tables** **7**

CHAPTER 1

# jwalk

jwalk performs random walks on a graph and learns representations for nodes using Word2Vec. It also has options to train existing models online and specify weights.

## Install

```
pip install -U jwalk
```

## Build

```
make build
```

## Usage

```
jwalk -i tests/data/karate.edgelist -o karate.emb --delimiter=' '
```

To see the full list of options:

```
jwalk --help

Prompt parameters:
  debug:            drop a debugger if an exception is raised
  delimiter:        delimiter for input file
  embedding-size:   dimension of word2vec embedding (default=200)
  has-header:       boolean if csv has header row
  help (-h):        argparse help
  input (-i):       file input (edgelist of 2/3 cols or adjacency matrix)
  log-level (-l)    logging level (default=INFO)
```

```
  model (-m):        use a pre-existing model
  num-walks (-n):    number of of random walks per graph (default=1)
  output (-o):       file output
  stats:             boolean to calculate walk statistics [requires pandas]
  undirected:        make graph undirected
  walk-length:       length of random walks (default=10)
  window-size:       word2vec window size (default=5)
  workers:           number of workers (default=multiprocessing.cpu_count)
```

### Input File

The input file can be of the following formats:

- Edgelist: CSV with 2 or 3 columns denoting the source, target and (optional) weight. There are CLI options to specify the delimiter and whether the file has a header (default=False). The CSV file is loaded using numpy if pandas is not installed. We strongly recommend using pandas to load the CSV as it's a lot faster.
- Graph: If the file has an extension that is ".npz", jwalk will assume that it is a SciPy CSR matrix. Included must be keys of data, indices, indptr, shape and labels (default=None) where labels are the node labels. For an example, see tests/data/karate.npz.

## Test

Running unit tests:

```
make test
```

Running linter:

```
make lint
```

Running tox:

```
make test-all
```

## Blog

Read more about jwalk in our blog post here: https://www.jwplayer.com/blog/deepwalk-recommendations/

## License

Apache License 2.0

## References


- [paper]: arXiv:1403.6652 [cs.SI] "DeepWalk: Online Learning of Social Representations"
- [paper]: arXiv:1607.00653 [cs.SI] "node2vec: Scalable Feature Learning for Networks"

CHAPTER 2

# Changelog

## Releases

### v0.5.0 (2017-01-10)

- First upload to PyPI.

CHAPTER 3

# Indices and tables

- genindex
- modindex
- search