
deepatari Documentation

Release

Ruben Glatt

July 29, 2016

1	Help	3
1.1	Installation guide	3
2	API reference	5
2.1	Experiment Classes	6
2.2	Agent Classes	6
2.3	Environment Classes	6
2.4	Learner Classes	6
2.5	Memory Classes	6
2.6	Statistics	6
3	About	7
3.1	License	7
3.2	Developers	7
3.3	Changelog	7
4	Indices and tables	9

This is a framework to support the development of algorithms for intelligent agents in the Atari game playing domain. The idea is to let researcher or programmer focus on the development of algorithms without having to think about the whole environment implementation and to facilitate and generalize the evaluation of these algorithms to make results comparable.

1.1 Installation guide

This guide will help install all the necessary software in a virtual environment generated with Anaconda. It is a step by step guide which might not be necessary for everyone. If you want to integrate the dependencies into your existing installation please do so.

Please be aware that I do not cover the installation of the Nvidia CUDA and CuDNN driver, please follow these instructions:

<http://docs.nvidia.com/cuda/cuda-getting-started-guide-for-linux/>

1.1.1 Create a working directory

Create a folder, where you download all necessary stuff and keep the rest of your system clean:

```
mkdir deepatari_stuff && cd deepatari_stuff
```

1.1.2 Prepare conda environment

If you haven't done it already, install anaconda with instructions from here:

<http://conda.pydata.org/docs/installation.html>

Create a conda environment with the basic packages and activate it:

```
conda create --name deepatariEnv python=2.7 pip numpy opencv matplotlib  
source activate deepatariEnv
```

1.1.3 Add non standard packages

The rest of the packages should be installed in this order, because they downgrade certain packages which will be upgraded later on again.

Download and install Neon [3]:

```
git clone https://github.com/NervanaSystems/neon.git  
cd neon && make sysinstall && cd ..
```

Download and install OpenAI gym [4]:

```
git clone https://github.com/openai/gym.git
cd gym && pip install -e '[atari]' && cd ..
```

Install the latest Theano [6] version :

```
pip install --upgrade https://github.com/Theano/Theano/archive/master.zip
```

Install the latest Lasagne [2] for Theano :

```
pip install --upgrade https://github.com/Lasagne/Lasagne/archive/master.zip
```

Install Tensorflow (Ubuntu/Linux 64-bit, GPU enabled. Requires CUDA toolkit 7.5 and CuDNN v4. Other options on library page.) [5]:

```
pip install --upgrade https://storage.googleapis.com/tensorflow/linux/gpu/tensorflow-0.8.0-cp27-none-
```

Install Keras [1]:

```
pip install keras
```

1.1.4 Install deepatari

The setup routine installs some runnable scripts to use directly from the command line:

```
git clone git@github.com:cowhi/deepatari.git
cd deepatari && python setup.py install && cd ..
```

Now you should be able to test the installation.

2.1 Experiment Classes

2.1.1 Experiment - Base class

2.1.2 AtariExp

2.2 Agent Classes

2.2.1 Agent - Base class

2.2.2 AtariAgent

2.3 Environment Classes

2.3.1 Environment - Base class

2.3.2 AtariEnv

2.4 Learner Classes

2.4.1 Learner - Base class

2.4.2 DQNNeon

2.4.3 DQNLasagne

2.5 Memory Classes

2.5.1 Memory - Base class

2.5.2 ReplayMemory

2.6 Statistics

3.1 License

```
The MIT License (MIT)
Copyright (c) 2016 Ruben Glatt
```

```
Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:
```

```
The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
```

3.2 Developers

- Ruben Glatt <ruben.glatt@gmail.com>

3.3 Changelog

3.3.1 Version 0.1

- Added documentation

3.3.2 Version 0.0

- Initial commit

Indices and tables

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