
proton*decaystudyDocumentation*

Release 0.1.0

Kevin Wierman

Feb 25, 2018

Contents

1	Installation	3
2	Usage	5
3	proton_decay_study	7
3.1	proton_decay_study package	7
4	Contributions	13
5	Indices and tables	15
	Python Module Index	17

Liquid argon time projection chambers (LArTPCs) are an innovative technology used in neutrino physics measurements that can also be utilized in establishing lifetimes on several partial lifetimes for proton and neutron decay. Current analyses suffer from low efficiencies and purities that arise from the misidentification of nucleon decay final states as background processes and vice-versa. One solution is to utilize convolutional neural networks (CNNs) to identify decay topologies in LArTPC data. In this study, CNNs are trained on Monte Carlo simulated data, labeled by truth, and then assessed by out-of-sample simulation. Currently running LArTPCs play an instrumental role in establishing the capabilities of this technology. Simultaneously, the next generation tens-of-kilotons flagship LArTPC experiment – one of whose main charges is to search for nucleon decay – is planning on using this technology in the future. We discuss analysis possibilities and further, a potential application of proton decay-sensitive CNN-enabled data acquisition.

Contents:

CHAPTER 1

Installation

As far as pre-requisites go, the big one is Tensorflow. We require *Tensorflow* $\geq 1.0.0$, but this is not strictly enforced due to some issues on certain machines.

The easiest way to install this package is by using pip:

```
pip install git+https://github.com/HEP-DL/proton_decay_study
```

If there is an existing installation, you can provide an upgrade flag:

```
pip install git+https://github.com/HEP-DL/proton_decay_study
```


CHAPTER 2

Usage

Nominally, this should be used via the Python API.

There are a few convenience endpoint definitions in *cli.py*.

For instance, the Kevnet training can be called with

```
train_kevnet --steps=100 --epochs=1000 --history=stage1.json --output=stage1.h5 dl_  
↪data/v04_00_00/*.h5
```

The other endpoints can be called similarly

3.1 proton_decay_study package

3.1.1 Subpackages

proton_decay_study.callbacks package

Submodules

proton_decay_study.callbacks.default module

Defines the default callbacks for usage in the mod:*proton_decay_study*

```
class proton_decay_study.callbacks.default.HistoryRecord
    Bases: keras.callbacks.Callback
    This is a stub in place for working on recording the training history
    on_batch_end (batch, logs={})
    on_train_begin (logs={})
    write (filename)
```

Module contents

proton_decay_study.config package

Module contents

```
class proton_decay_study.config.Config
    Bases: object
```

Represents configuration of network training

proton_decay_study.generators package

Submodules

proton_decay_study.generators.base module

class proton_decay_study.generators.base.**BaseDataGenerator**

Bases: object

Base data generator which hooks into the networks to provide an interface to the incoming data.

logger = <logging.Logger object>

next ()

proton_decay_study.generators.gen3d module

class proton_decay_study.generators.gen3d.**Gen3D** (datapaths, datasetname, labelsetname, batch_size=1)

Bases: *proton_decay_study.generators.base.BaseDataGenerator*

Creates a generator for a list of files

input

Input shape property

Returns: A tuple representing

logger = <logging.Logger object>

next ()

output

Output shape property

Returns: A tuple representing the shape of the first data this picks out of the file

class proton_decay_study.generators.gen3d.**Gen3DRandom** (datapaths, datasetname, labelsetname, batch_size=1)

Bases: *proton_decay_study.generators.gen3d.Gen3D*

proton_decay_study.generators.multi_file module

class proton_decay_study.generators.multi_file.**MultiFileDataGenerator** (datapaths, datasetname, labelsetname, batch_size=10)

Bases: *proton_decay_study.generators.base.BaseDataGenerator*

Creates a generator for a list of files

input

```
logger = <logging.Logger object>
next ()
    This should iterate over both files and datasets within a file.
output
```

proton_decay_study.generators.single_file module

```
class proton_decay_study.generators.single_file.SingleFileDataGenerator (datapath,
                                                                           dataset,
                                                                           la-
                                                                           belset,
                                                                           batch_size=10)

Bases: proton_decay_study.generators.base.BaseDataGenerator
Creates a generator for a single file

logger = <logging.Logger object>
next ()
```

proton_decay_study.generators.threaded_gen3d module

```
class proton_decay_study.generators.threaded_gen3d.SingleFileThread (datasetname,
                                                                        labelset-
                                                                        name,
                                                                        batch_size)

Bases: threading.Thread
Wrapper thread for buffering data from a single file

activeThreads = []

static killRunThreads (frame)
    Sets the thread kill flag to each of the ongoing analysis threads

logger = <logging.Logger object>
queue = <Queue.Queue instance>
queueLock = <thread.lock object>

run ()
    Loops over queue to accept new configurations

single_status ()

static startThreads (datasetname, labelsetname, batch_size)

static status ()

threadLock = <thread.lock object>

visit (parent)

static waitTillComplete ()
```

```
class proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator (datapaths,  
dataset-  
name,  
la-  
belset-  
name,  
batch_size=  
nThreads=8
```

Bases: `proton_decay_study.generators.base.BaseDataGenerator`

Uses threads to pull asynchronously from files

`check_and_refill()`

`input`

`logger = <logging.Logger object>`

`next()`

`output`

`status()`

proton_decay_study.generators.threaded_multi_file module

```
class proton_decay_study.generators.threaded_multi_file.SingleFileThread (datasetname,  
la-  
belset-  
name,  
batch_size)
```

Bases: `threading.Thread`

Represents a single file that is asynchronously

`activeThreads = []`

`get`

`static killRunThreads (frame)`

Sets the thread kill flag to each of the ongoing analysis threads

`logger = <logging.Logger object>`

`queue = <Queue.Queue instance>`

`queueLock = <thread.lock object>`

`run()`

Loops over queue to accept new configurations

`static startThreads (datasetname, labelsetname, batch_size)`

`threadLock = <thread.lock object>`

`static waitTillComplete()`

```
class proton_decay_study.generators.threaded_multi_file.ThreadedMultiFileDataGenerator (datap
    datas
    name
    la-
    belse
    name
    batch
    nThr

    Bases: proton_decay_study.generators.base.BaseDataGenerator
    Uses threads to pull asynchronously from files
    input
    logger = <logging.Logger object>
    next ()
    output
```

Module contents

proton_decay_study.models package

Submodules

proton_decay_study.models.kevnet module

```
class proton_decay_study.models.kevnet.Kevnet (generator)
    Bases: keras.engine.training.Model
    assemble (generator)
    logger = <logging.Logger object>
```

proton_decay_study.models.kevnet_fcn module

```
class proton_decay_study.models.kevnet_fcn.capetian_modifier (generator)
    Bases: proton_decay_study.models.kevnet.Kevnet
    logger = <logging.Logger object>
class proton_decay_study.models.kevnet_fcn.percussive_treashership (generator)
    Bases: proton_decay_study.models.kevnet.Kevnet
    logger = <logging.Logger object>
```

Module contents

proton_decay_study.visualization package

Submodules

proton_decay_study.visualization.intermediate module

```
class proton_decay_study.visualization.intermediate.IntermediateVisualizer (model,
                                                                    layer_name,
                                                                    data)
    Bases: keras.engine.training.Model
    infer ()
```

proton_decay_study.visualization.kevnet module

```
class proton_decay_study.visualization.kevnet.KevNetVisualizer (model, data)

    initialize ()
    layers = ['block1_conv1', 'block2_conv1', 'block3_conv1', 'block4_conv1', 'block5_conv1']
    logger = <logging.Logger object>
    makedirs ()
    run ()
```

Module contents

3.1.2 Submodules

3.1.3 proton_decay_study.cli module

3.1.4 Module contents

Base package for Proton Decay Study :author: Kevin Wierman

CHAPTER 4

Contributions

Your contributions to this are more than welcome!

CHAPTER 5

Indices and tables

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

p

- `proton_decay_study`, [12](#)
- `proton_decay_study.callbacks`, [7](#)
- `proton_decay_study.callbacks.default`, [7](#)
- `proton_decay_study.cli`, [12](#)
- `proton_decay_study.config`, [7](#)
- `proton_decay_study.generators`, [11](#)
- `proton_decay_study.generators.base`, [8](#)
- `proton_decay_study.generators.gen3d`, [8](#)
- `proton_decay_study.generators.multi_file`,
[8](#)
- `proton_decay_study.generators.single_file`,
[9](#)
- `proton_decay_study.generators.threaded_gen3d`,
[9](#)
- `proton_decay_study.generators.threaded_multi_file`,
[10](#)
- `proton_decay_study.models`, [12](#)
- `proton_decay_study.models.kevnet`, [11](#)
- `proton_decay_study.models.kevnet_fcn`,
[11](#)
- `proton_decay_study.visualization`, [12](#)
- `proton_decay_study.visualization.intermediate`,
[12](#)
- `proton_decay_study.visualization.kevnet`,
[12](#)

A

activeThreads (proton_decay_study.generators.threaded_gen3d.SingleFileThread attribute), 9

activeThreads (proton_decay_study.generators.threaded_multi_file.SingleFileThread attribute), 10

assemble() (proton_decay_study.models.kevnet.Kevnet method), 11

input (proton_decay_study.generators.multi_file.MultiFileDataGenerator attribute), 8

input (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator attribute), 10

input (proton_decay_study.generators.threaded_multi_file.ThreadedMultiFileDataGenerator attribute), 11

IntermediateVisualizer (class in proton_decay_study.visualization.intermediate), 12

B

BaseDataGenerator (class in proton_decay_study.generators.base), 8

C

capetian_modifier (class in proton_decay_study.models.kevnet_fcn), 11

check_and_refill() (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator method), 10

Config (class in proton_decay_study.config), 7

G

Gen3D (class in proton_decay_study.generators.gen3d), 8

Gen3DRandom (class in proton_decay_study.generators.gen3d), 8

get (proton_decay_study.generators.threaded_multi_file.SingleFileThread attribute), 10

H

HistoryRecord (class in proton_decay_study.callbacks.default), 7

I

infer() (proton_decay_study.visualization.intermediate.IntermediateVisualizer method), 12

initialize() (proton_decay_study.visualization.kevnet.KevNetVisualizer method), 12

input (proton_decay_study.generators.gen3d.Gen3D attribute), 8

K

Kevnet (class in proton_decay_study.models.kevnet), 11

KevNetVisualizer (class in proton_decay_study.visualization.kevnet), 12

killRunThreads() (proton_decay_study.generators.threaded_gen3d.SingleFileThread static method), 9

killRunThreads() (proton_decay_study.generators.threaded_multi_file.SingleFileThread static method), 10

L

layers (proton_decay_study.visualization.kevnet.KevNetVisualizer attribute), 12

logger (proton_decay_study.generators.base.BaseDataGenerator attribute), 8

logger (proton_decay_study.generators.gen3d.Gen3D attribute), 8

logger (proton_decay_study.generators.multi_file.MultiFileDataGenerator attribute), 8

logger (proton_decay_study.generators.single_file.SingleFileDataGenerator attribute), 9

logger (proton_decay_study.generators.threaded_gen3d.SingleFileThread attribute), 9

logger (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator attribute), 10

logger (proton_decay_study.generators.threaded_multi_file.SingleFileThread attribute), 10

logger (proton_decay_study.generators.threaded_multi_file.ThreadedMultiFileDataGenerator attribute), 11

logger (proton_decay_study.models.kevnet.Kevnet attribute), 11

logger (proton_decay_study.models.kevnet_fcn.capetian_model attribute), 11
 logger (proton_decay_study.models.kevnet_fcn.percussive_treasure attribute), 11
 logger (proton_decay_study.visualization.kevnet.KevNetVisualizer attribute), 12

M

mkdir() (proton_decay_study.visualization.kevnet.KevNetVisualizer method), 12
 MultiFileDataGenerator (class in proton_decay_study.generators.multi_file), 8

N

next() (proton_decay_study.generators.base.BaseDataGenerator method), 8
 next() (proton_decay_study.generators.gen3d.Gen3D method), 8
 next() (proton_decay_study.generators.multi_file.MultiFileDataGenerator method), 9
 next() (proton_decay_study.generators.single_file.SingleFileDataGenerator method), 9
 next() (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator method), 10
 next() (proton_decay_study.generators.threaded_multi_file.ThreadedMultiFileDataGenerator method), 11

O

on_batch_end() (proton_decay_study.callbacks.default.HistoryRecord method), 7
 on_train_begin() (proton_decay_study.callbacks.default.HistoryRecord method), 7
 output (proton_decay_study.generators.gen3d.Gen3D attribute), 8
 output (proton_decay_study.generators.multi_file.MultiFileDataGenerator attribute), 9
 output (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileDataGenerator attribute), 10
 output (proton_decay_study.generators.threaded_multi_file.ThreadedMultiFileDataGenerator attribute), 11

P

percussive_treasure (class in proton_decay_study.models.kevnet_fcn), 11
 proton_decay_study (module), 12
 proton_decay_study.callbacks (module), 7
 proton_decay_study.callbacks.default (module), 7
 proton_decay_study.cli (module), 12
 proton_decay_study.config (module), 7
 proton_decay_study.generators (module), 11
 proton_decay_study.generators.base (module), 8
 proton_decay_study.generators.gen3d (module), 8
 proton_decay_study.generators.multi_file (module), 8

proton_decay_study.generators.single_file (module), 9
 proton_decay_study.generators.threaded_gen3d (module), 9
 proton_decay_study.generators.threaded_multi_file (module), 10
 proton_decay_study.models (module), 12
 proton_decay_study.models.kevnet (module), 11
 proton_decay_study.models.kevnet_fcn (module), 11
 proton_decay_study.visualization (module), 12
 proton_decay_study.visualization.intermediate (module), 12
 proton_decay_study.visualization.kevnet (module), 12

Q

queue (proton_decay_study.generators.threaded_gen3d.SingleFileThread attribute), 9
 queue (proton_decay_study.generators.threaded_multi_file.SingleFileThread attribute), 10
 queue_lock (proton_decay_study.generators.threaded_gen3d.SingleFileThread attribute), 9
 queue_lock (proton_decay_study.generators.threaded_multi_file.SingleFileThread attribute), 10

R

run() (proton_decay_study.generators.threaded_gen3d.SingleFileThread method), 9
 run() (proton_decay_study.generators.threaded_multi_file.SingleFileThread method), 10
 run() (proton_decay_study.visualization.kevnet.KevNetVisualizer method), 12

S

single_status() (proton_decay_study.generators.threaded_gen3d.SingleFileThread attribute), 9
 SingleFileDataGenerator (class in proton_decay_study.generators.single_file), 9
 SingleFileThread (class in proton_decay_study.generators.threaded_gen3d), 9
 SingleFileThread (class in proton_decay_study.generators.threaded_multi_file), 10
 startThreads() (proton_decay_study.generators.threaded_gen3d.SingleFileThread static method), 9
 startThreads() (proton_decay_study.generators.threaded_multi_file.SingleFileThread static method), 10
 status() (proton_decay_study.generators.threaded_gen3d.SingleFileThread static method), 9
 status() (proton_decay_study.generators.threaded_gen3d.ThreadedMultiFileThread static method), 10

T

ThreadedMultiFileDataGenerator (class in proton_decay_study.generators.threaded_multi_file), 10

```

        ton_decay_study.generators.threaded_gen3d),
9
ThreadedMultiFileDataGenerator (class in pro-
    ton_decay_study.generators.threaded_multi_file),
10
threadLock (proton_decay_study.generators.threaded_gen3d.SingleFileThread
    attribute), 9
threadLock (proton_decay_study.generators.threaded_multi_file.SingleFileThread
    attribute), 10

```

V

```

visit() (proton_decay_study.generators.threaded_gen3d.SingleFileThread
    method), 9

```

W

```

waitTillComplete() (pro-
    ton_decay_study.generators.threaded_gen3d.SingleFileThread
    static method), 9
waitTillComplete() (pro-
    ton_decay_study.generators.threaded_multi_file.SingleFileThread
    static method), 10
write() (proton_decay_study.callbacks.default.HistoryRecord
    method), 7

```