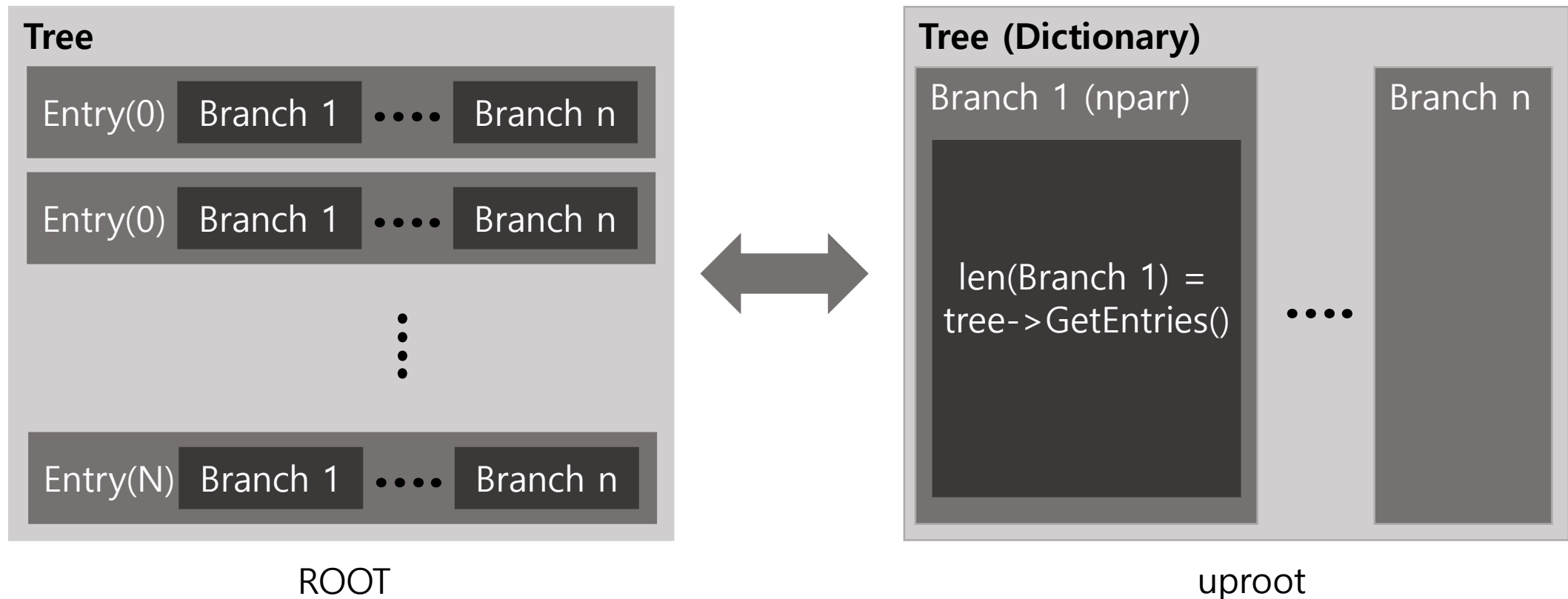


uproot

- uproot is ROOT I/O module based on pure Python.
- No ROOT infrastructure is needed.
- ROOT file can be converted into other format frequently used in python environment.
- To reduce loaded data on RAM, it can read file within the limit set by user.

uproot data structure

- Column like structure (saying ROOT tree has row like structure)
- Tree : Dictionary -> keys are branch IDs
- Branch : NumpyArray(jagged)



ROOT file reading test

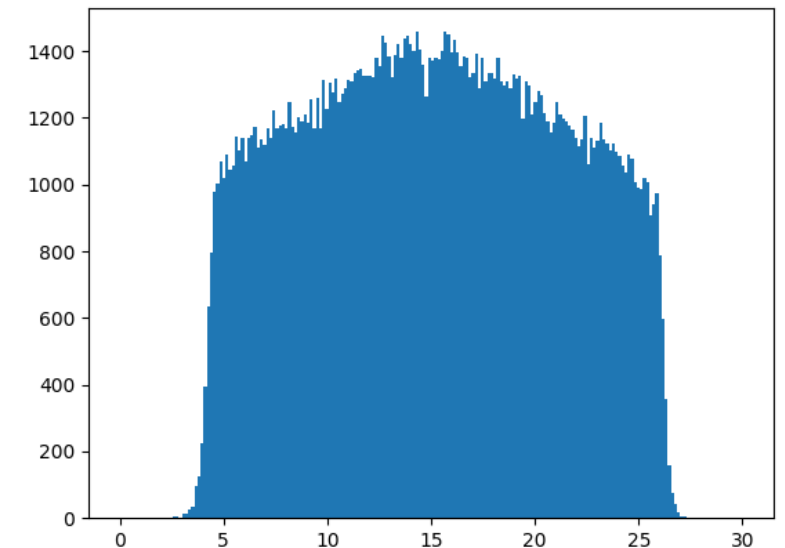
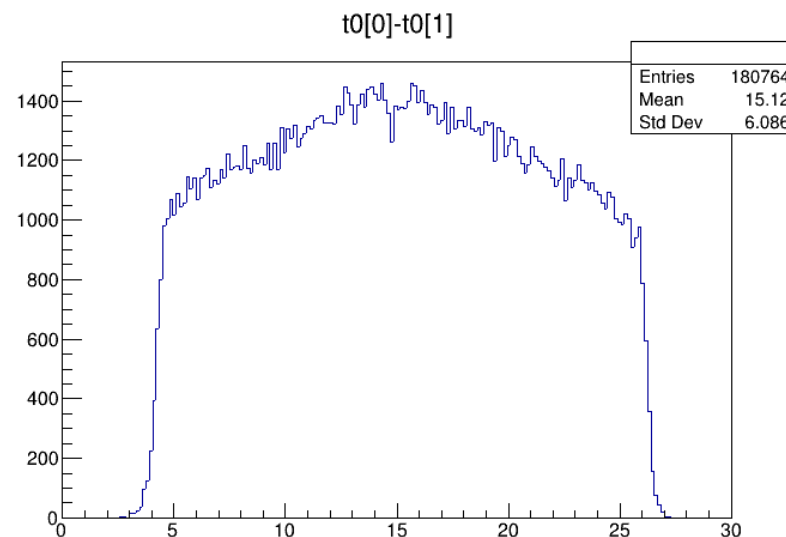
- Cosmic plastic scintillator hit data is used.
- Histogram of (Left PMT time – Right PMT time)
- Histogram is drawn by using matplotlib module.

```
hblee@localhost:~/data/coding/UProot
File Edit View Search Terminal Help
from tqdm import tqdm
import uproot
import matplotlib.pyplot as plt

file = uproot.open("~/data/trigsim/v4/RQ0YT004_20191126_evrec.root")
tree = file["EvRec"] #tree is dictionary
br = tree.arrays(namedecode='utf=8') # branches are numpy array
ar = []

for i in tqdm(range(len(br['t0'])), desc='Progress'):
    ar.append(br['t0'][i][0]-br['t0'][i][1])

print (len(ar))
plt.hist(ar,bins=200,range=(0,30))
plt.show()
```



ROOT file read and feed to Keras

- This code is at <https://github.com/physmlee/DLStudy/uprootMNIST>
- Seungmok made MNIST.root in order to test TMVA.
- Test importing uproot to read root file and making numpy array for machine running in the pure python environment.
- Tree structure is as follows.

ROOT file read and feed to Keras

```
for i in tqdm(range(10), desc='Load Tree'):
    TrainTrlist.append(file['train%d' % (i)])
    TestTrlist.append(file['test%d' % (i)])
    TrainBrancheslist.append(TrainTrlist[i].arrays(namedecode='utf-8'))
    TestBrancheslist.append(TestTrlist[i].arrays(namedecode='utf-8'))
```

```
BranchesName = TrainBrancheslist[0].keys() #Get Branch names
```

```
for j in tqdm(range(10), desc='Load train Data'):
    for i in range(len(TrainBrancheslist[j][BranchesName[0]])):
        arr = []
        for BranchName in BranchesName: #reading a row in the tree
            arr.append(TrainBrancheslist[j][BranchName][i])
        xtrainarr.append(arr)
        ytrainarr.append(j)
```

784 entry array (an image)

60000 images

xtrainarr = [[---], [---], [---], - , [---]]

ytrainarr = [-, -, -, -, -]

Corresponding image class (0-9)

j

Train(test) tree(image of handwritten "0")

Entry 0

BranchName

Branch0 (0th pixel of image)

Branch1 (1st pixel of image)

...

Branch783 (783rd pixel of image)

:

i

Entry N_0

Branch0 (0th pixel of image)

Branch1 (1st pixel of image)

...

Branch783 (783rd pixel of image)

⋮

Train(test) tree(image of handwritten "9")

Entry 0

Branch0 (0th pixel of image)

Branch1 (1st pixel of image)

...

Branch783 (783rd pixel of image)

:

Entry N_9

Branch0 (0th pixel of image)

Branch1 (1st pixel of image)

...

Branch783 (783rd pixel of image)

File Edit View Search Terminal Help

```

[hblee@localhost uprootMNIST]$ vim MNISTrootload.py
[hblee@localhost uprootMNIST]$ python MNISTrootload.py
Using TensorFlow backend.
Load Tree: 100%|██████████| 10/10 [00:06<00:00, 1.66it/s]
Load train Data: 100%|██████████| 10/10 [00:12<00:00, 1.28s/it]
Load test Data: 100%|██████████| 10/10 [00:02<00:00, 3.67it/s]
WARNING:tensorflow:From /home/hblee/.local/lib/python2.7/site-packages/tensorflow_core/python/ops/resource_variable_ops.py:1630: calling __init__ (from tensorflow.python.ops.resource_variable_ops) with constraint is deprecated and will be removed in a future version.
Instructions for updating:
If using Keras pass *_constraint arguments to layers.
2020-02-10 14:13:37.237520: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcuda.so.1'; dlerror: libcuda.so.1: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /home/hblee/proc/Geant4/lib64:/home/hblee/proc/rootv2/lib:/root:/home/hblee/proc/Geant4/lib64:/home/hblee/proc/rootv2/lib:/root:/home/hblee/proc/CLHEP/2.4.1.0/lib:/usr/lib64:/home/hblee/proc/CLHEP/2.4.1.0/lib:/usr/lib64
2020-02-10 14:13:37.237857: E tensorflow/stream_executor/cuda/cuda_driver.cc:318] failed call to cuInit: UNKNOWN ERROR (303)
2020-02-10 14:13:37.237891: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] kernel driver does not appear to be running on this host (localhost.localdomain): /proc/driver/nvidia/version does not exist
2020-02-10 14:13:37.238218: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-02-10 14:13:37.249951: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU Frequency: 1992000000 Hz
2020-02-10 14:13:37.251188: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x34eb05c0 initialized for platform Host (this does not guarantee that XLA will be used). Devices:
2020-02-10 14:13:37.251209: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
WARNING:tensorflow:From /home/hblee/.local/lib/python2.7/site-packages/keras/backend/tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

Epoch 1/5
60000/60000 [=====] - 2s 42us/step - loss: 0.2666 - accuracy: 0.9252
Epoch 2/5
60000/60000 [=====] - 2s 40us/step - loss: 0.1084 - accuracy: 0.9685
Epoch 3/5
60000/60000 [=====] - 2s 39us/step - loss: 0.0712 - accuracy: 0.9790
Epoch 4/5
60000/60000 [=====] - 2s 39us/step - loss: 0.0514 - accuracy: 0.9851
Epoch 5/5
60000/60000 [=====] - 2s 38us/step - loss: 0.0377 - accuracy: 0.9892
10000/10000 [=====] - 0s 26us/step
('test_acc: ', 0.9797000288963318)
[hblee@localhost uprootMNIST]$

```

- The result is good.
- Memory consuming is big.