



SPRACE

# GraphSAGE and Lightning study - part I

JEFFERSON FIALHO

SPRACE-ML (track-ml)

# Lightning PyTorch

- Lightning is a API to build complex AI models
- With Lightning you can abstract the details of training
- Packages ready for:
  - template models
  - Run your code on any hardware
  - Performance & bottleneck profiler
  - Visualization
- “Spend more time on **research**, less on **engineering**”
- LightningModule = The main module

# Lightning PyTorch

## Examples:

Example:

```
import pytorch_lightning as pl
from pl_bolts.models.vision import ImageGPT

dm = MNISTDataModule('.')
model = ImageGPT(dm)

pl.Trainer(gpu=4).fit(model)
```

Convolutional Architectures

GPT-2

Image GPT

Pixel CNN

UNet

Semantic Segmentation

# Lightning PyTorch

Problem:

```
model_library: /data/jfialho/exatrnx/new/Tracking-ML-Exa.TrkX/src/Pipelines/TrackML_Example/LightningModules
artifact_library: /global/cscratch1/sd/danielm/ExaTrkX/lightning_checkpoints

model_list:
- {set: Filter, name: VanillaFilter, config: train_filter.yaml, resume_id: enm731r3, overwrite:
  {
    output_dir: /global/cscratch1/sd/danielm/ExaTrkX/trackml_processed/filter_processed/0 pt cut endcaps test,
    callbacks: [FilterInferenceCallback]
  }
}
```

# GraphSAGE

[GraphSAGE](#) is a framework for inductive representation learning on large graphs (graph sampling). GraphSAGE is used to **generate low-dimensional vector representations for nodes**, and is especially useful for **graphs that have rich node attribute information**.

- for large graphs that force us beyond the available memory of our GPU or CPU (graph sampling)
- graph sampling: NeighbourSampler, GraphSAINTSampler
- SAGEConv (from pytorch)

# GraphSAGE

