GraphNeT Overview

GraphNeT Workshop / 2 May 2023

Andreas Søgaard

Niels Bohr Institute, University of Copenhagen

Danish
Data Science
Academy

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 890778.

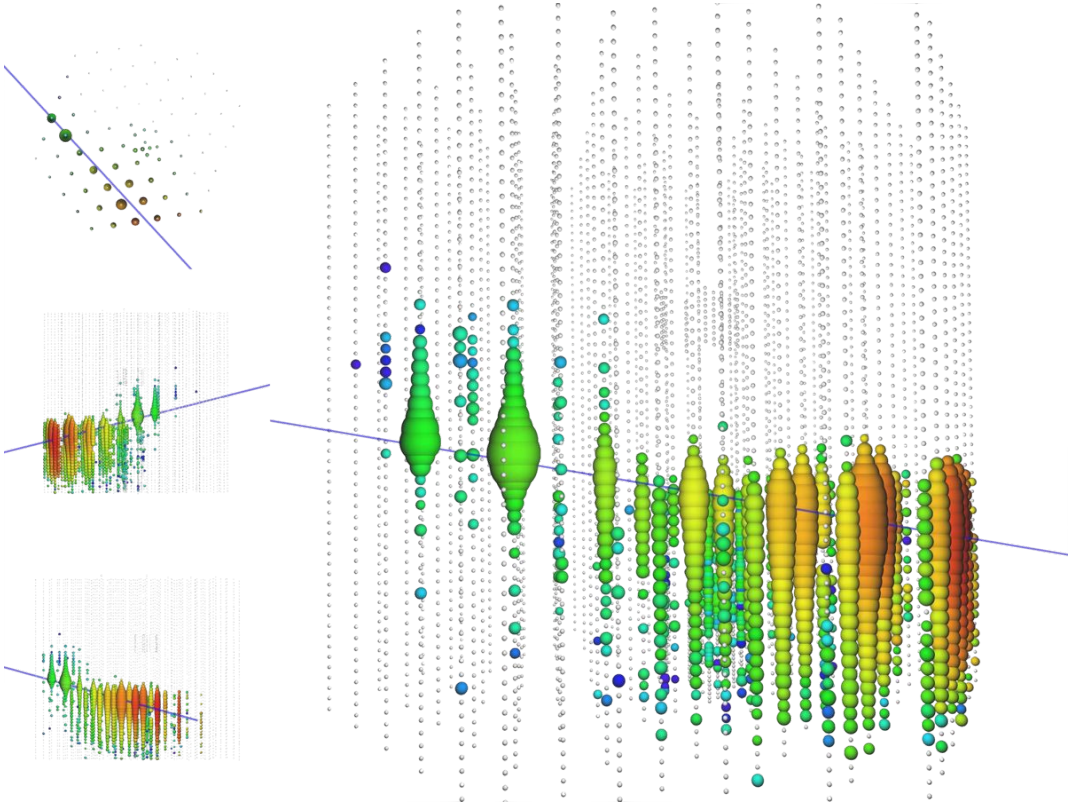
Outline

Background
Motivation for GraphNeT
Current status
First public release (v1.0)
Roadmap to v2.0

Background

Neutrino telescope data is unique

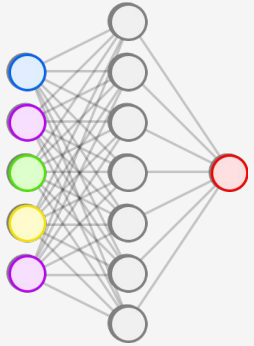
- High rates, O(kHz) × O(10 years)
- Often sparse charge deposits
- Non-trivial geometry
- Geometrically heterogeneous, nested sub-detectors
- Similar detection method in all sub-detectors
- Very time-consuming for traditional optimisation-based reconstruction methods



Common types of neural networks

(Deep) Neural network — (D)NN

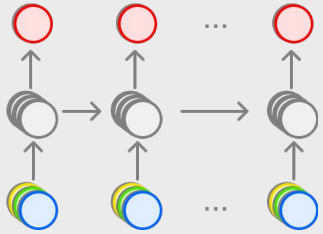
Occasionally: Multi-layer perceptron (MLP)



Structure
No inherent geometric structure

Example
Engineered, high-level event features

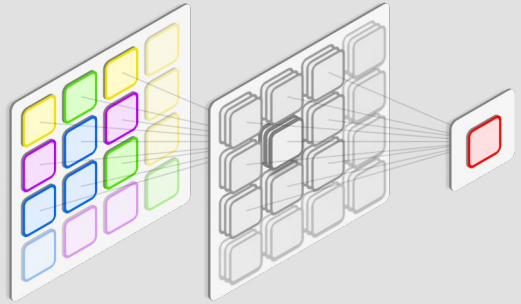
Recurrent neural network (RNN)



Structure
Sequential

Example
Time-series

Convolutional neural network (CNN)



Structure
Orthogonal data, translation invariance

Example
Images

Unifying “zoo” of architectures

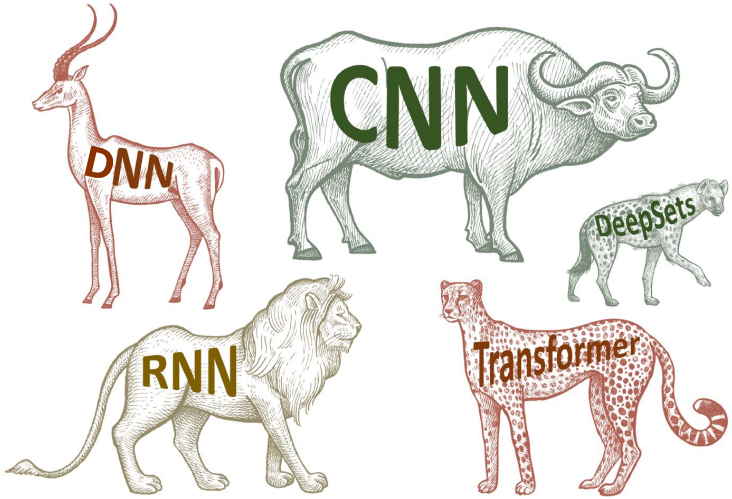
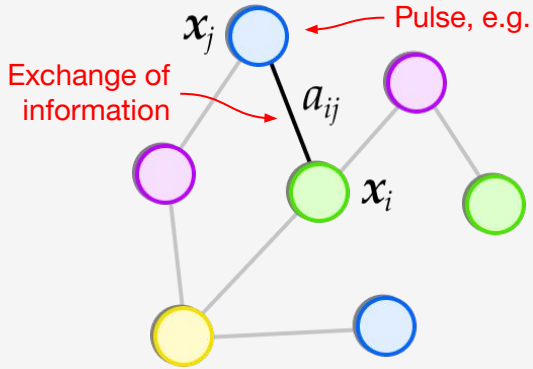


Image adapted from:
Bronstein, Geometric foundations of deep learning (2021)

Graph neural network — GNN

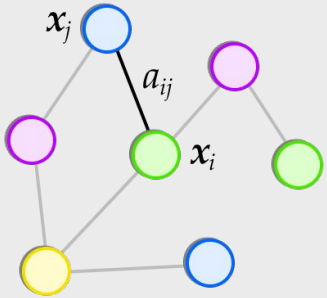
Representation
Data $\boldsymbol{x}_i$ on nodes a graph; nodes connected by edges $\boldsymbol{a}_{ij}$

Structure
Any that can be encoded through *adjacency* of nodes



Most neural network architectures can be seen as special cases of the GNN with added structure
(Bronstein et al., 2021)

Why GNNs are a natural choice in neutrino telescopes



Challenge

- ⛰️ Sparse charge deposits
- ⛰️ Non-trivial geometry
- ⛰️ Nested sub-detectors

Addressed

- ✅ Only ingest hit PMTs in each event
- ✅ No requirements on structure
- ✅ Information transfer among all nodes

Plus:

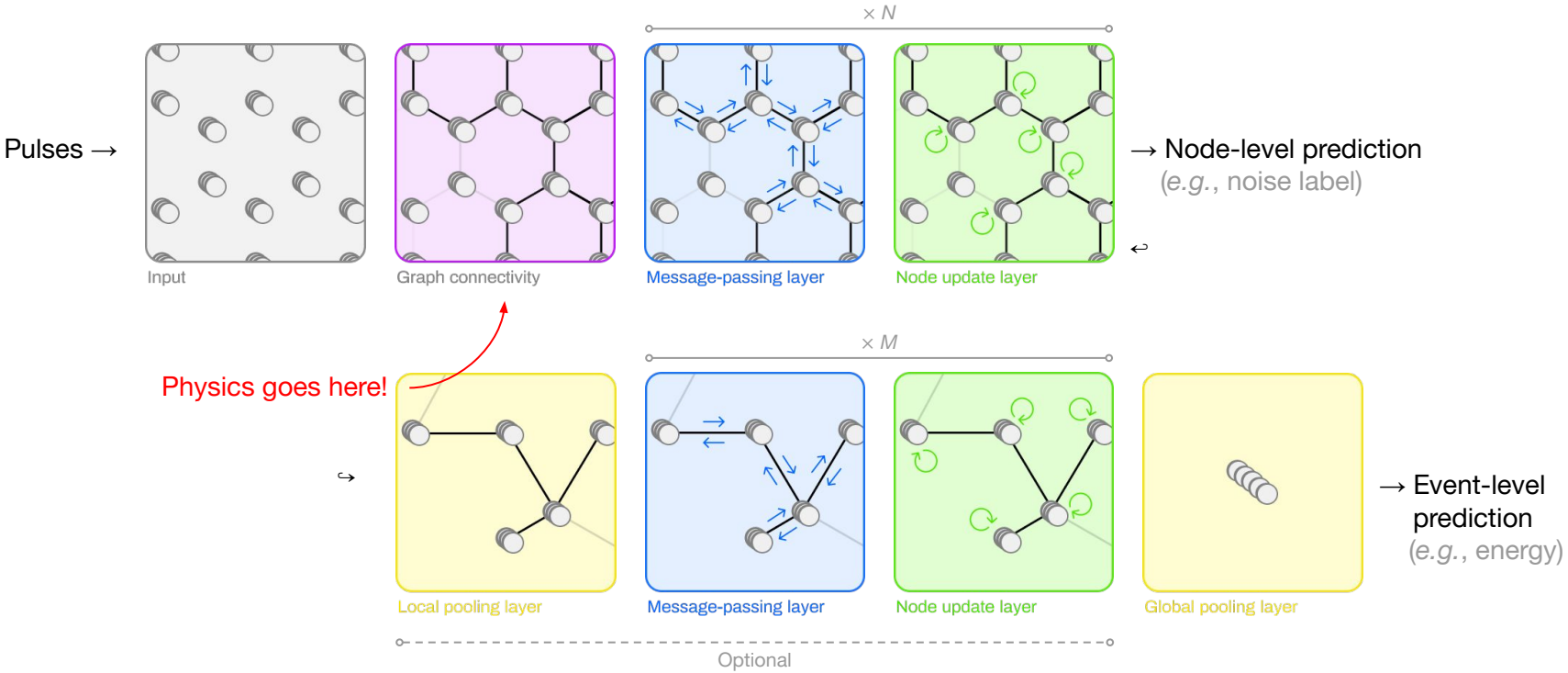
- ✅ No need to “summarise” pulses to DOM-level
- ✅ No need for specialised kernels
- ✅ Ability to *encode* physics, material properties, etc. into structure, e.g.,

$$a_{ij} \sim \frac{\text{sign}(\Delta t_{ij})}{1 + \| x_i^{\mu} - x_j^{\mu} \|^2}$$

… with the regular DL benefit:

- ✅ Fast inference

Anatomy of a graph neural network



Motivation for GraphNeT

The challenge with machine learning in physics

Challenge

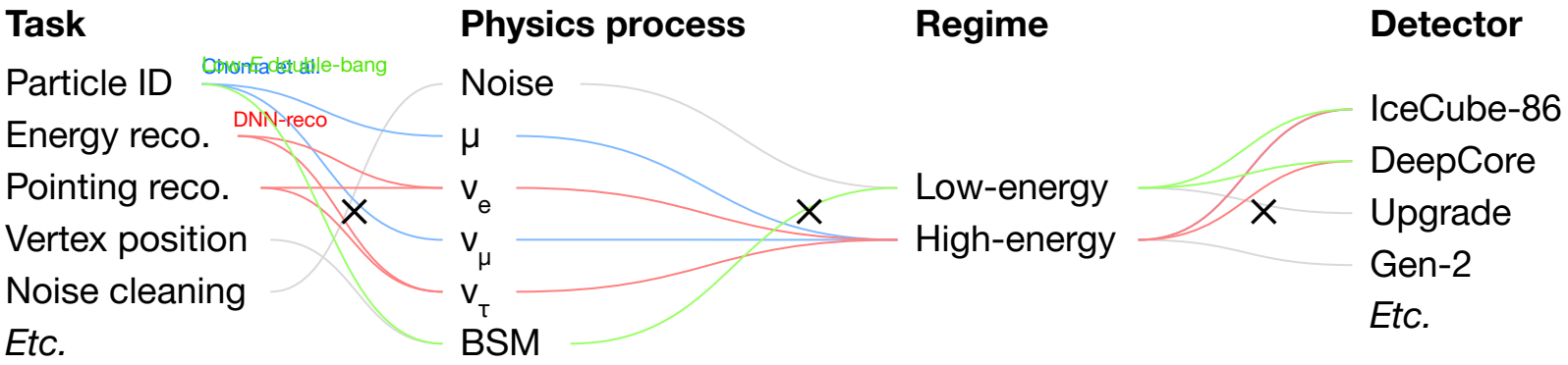
- ⛰️ Developers may have little ML experience
- ⛰️ Siloed development, often from scratch

Risk

- ❌ Brittle, suboptimal solutions
- ❌ Time spent on “boilerplate” instead of physics

We’re working on very related problems

Similar detectors, data, physics processes, deployment setting, end-users, etc.



“Zoo” of use cases not solved holistically

Potential for new ways of working with ML in physics

Potential

- Address “zoo” of ML use cases holistically
- Using validated, best-practices code
- Efficient software/ML development workflows

Outcome

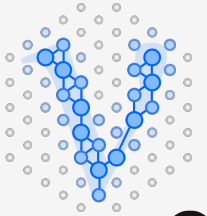
- More time for physics
- Better, more reliable results
- Contributions of individual ML developers has a broader, lasting impact in the collaboration

Proposition

A framework for developing GNN-based tools for neutrino telescopes (and beyond!)

One-stop-shop for model development to deployment:

- ML developers
  - Everything needed to build, train, and validate GNNs for physics
- Physics end-users
  - Can choose from a library of pre-trained models and apply them, e.g., as IceTray modules



GraphNeT

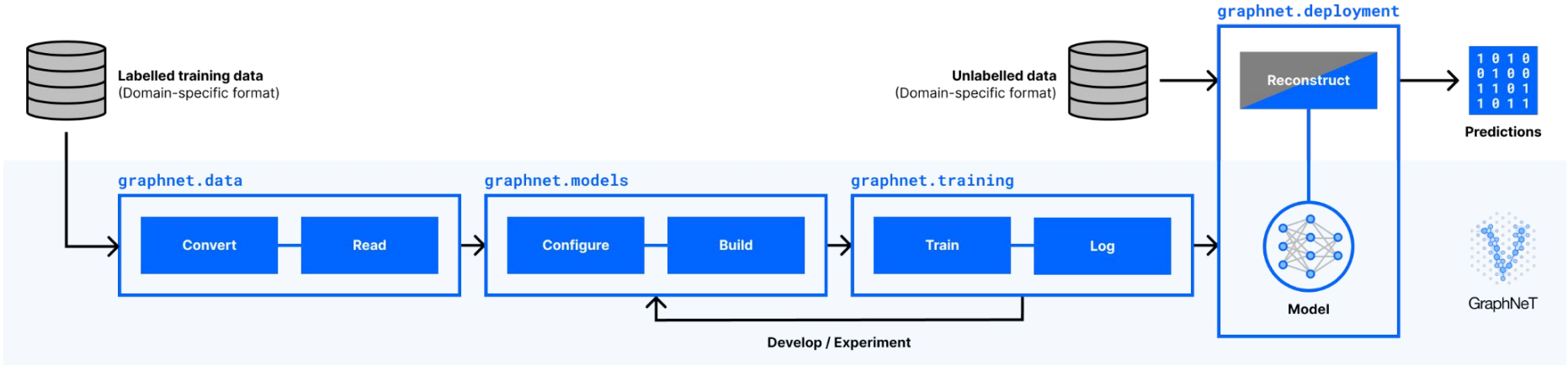
Graph Neural Networks for
Neutrino Telescope Event Reconstruction

graphnet-team/graphnet

graphnet-team

- Reusable GNN components for plug-and-play ML
- All components for end-to-end ML pipeline (data → prod.)
- Validated code, following best practices
- Applicable across all of IceCube **+ other experiments**

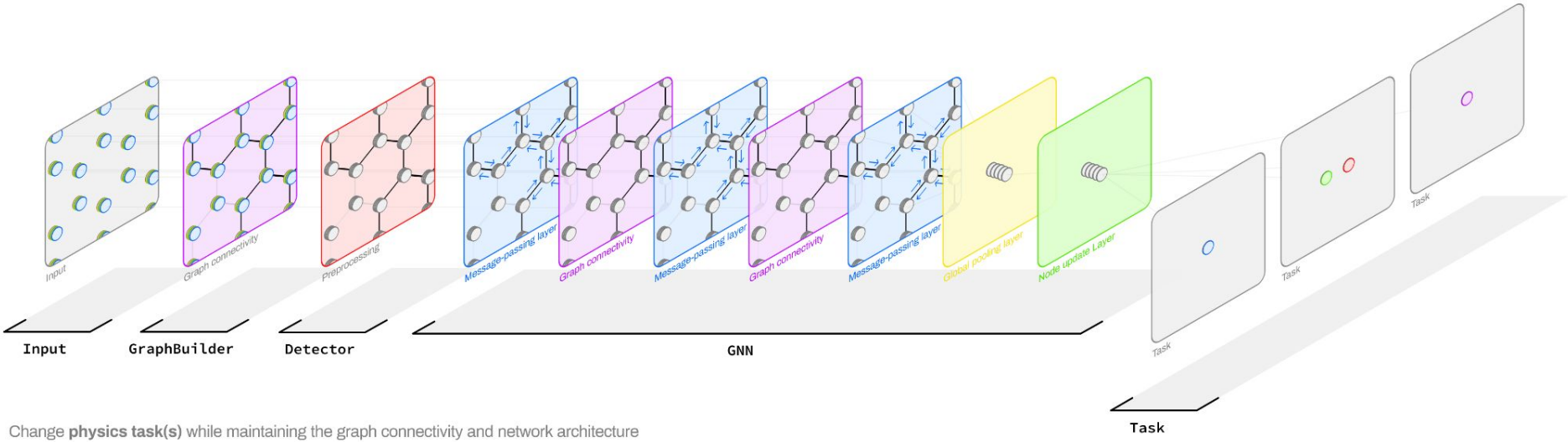
Factoring out ML from physics



Stack: PyTorch PyG PyTorch Lightning W&B docker

GraphNeT in a nutshell

Modularised, plug-and-play ML components for any use case.



Change **physics task(s)** while maintaining the graph connectivity and network architecture

GraphNeT in a nutshell

```python
from graphnet import (
  EuclideanGraphBuilder as GraphBuilder,
  IceCubeUpgrade as Detector,
  DynEdge_V2 as GNN,
  EnergyReconstruction as Task,
)

# Go do physics!
```

…ideally

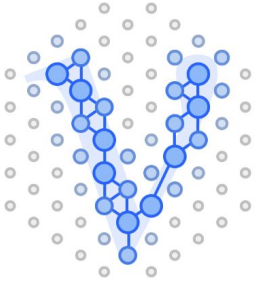
```python
from graphnet import Model

model = Model.from_pretrained(
    "icecube-low-energy-neutrino-v2"
)

# Go do physics!
```

Current status

README.md



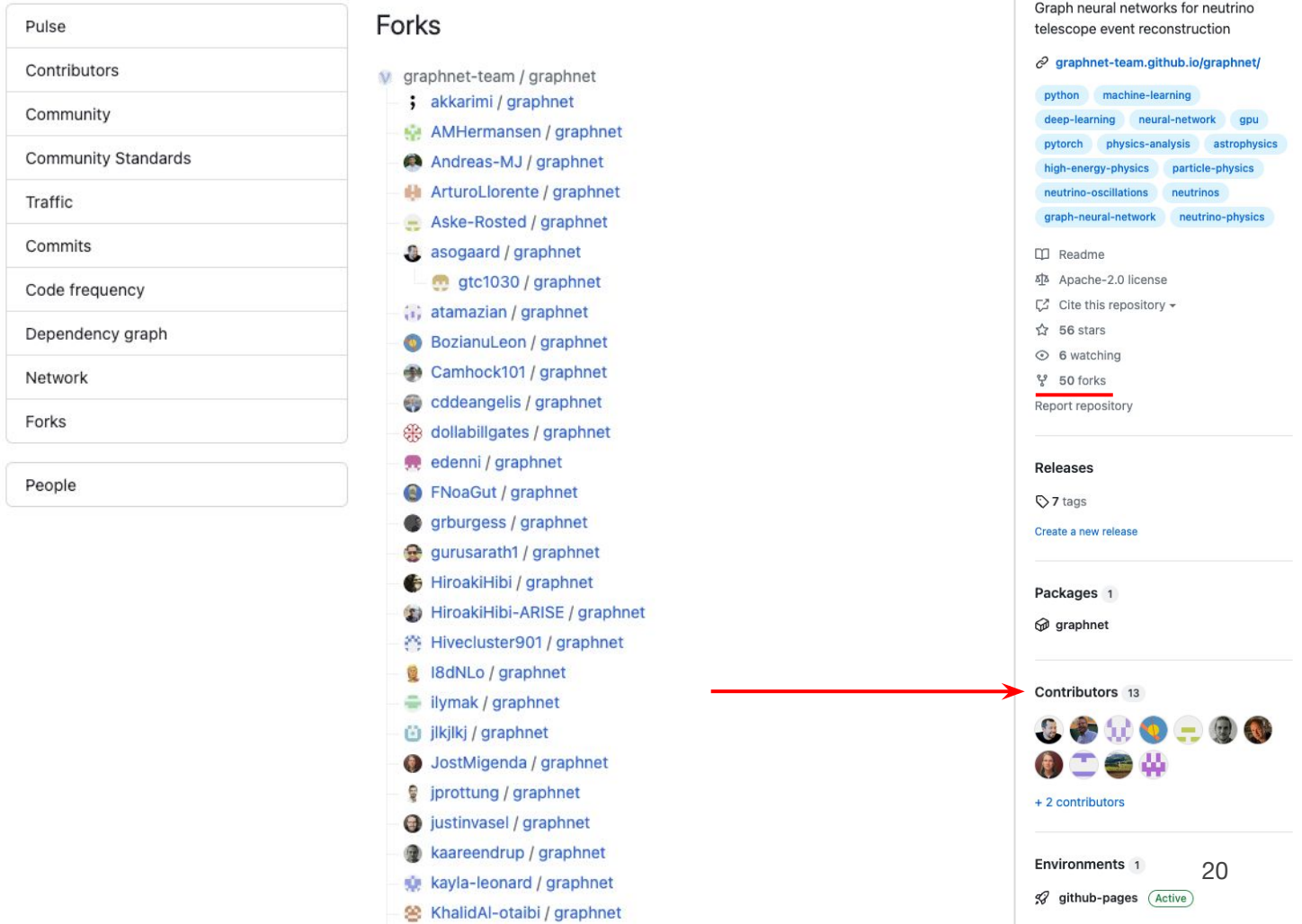
GraphNeT

Graph Neural Networks for
Neutrino Telescope Event Reconstruction

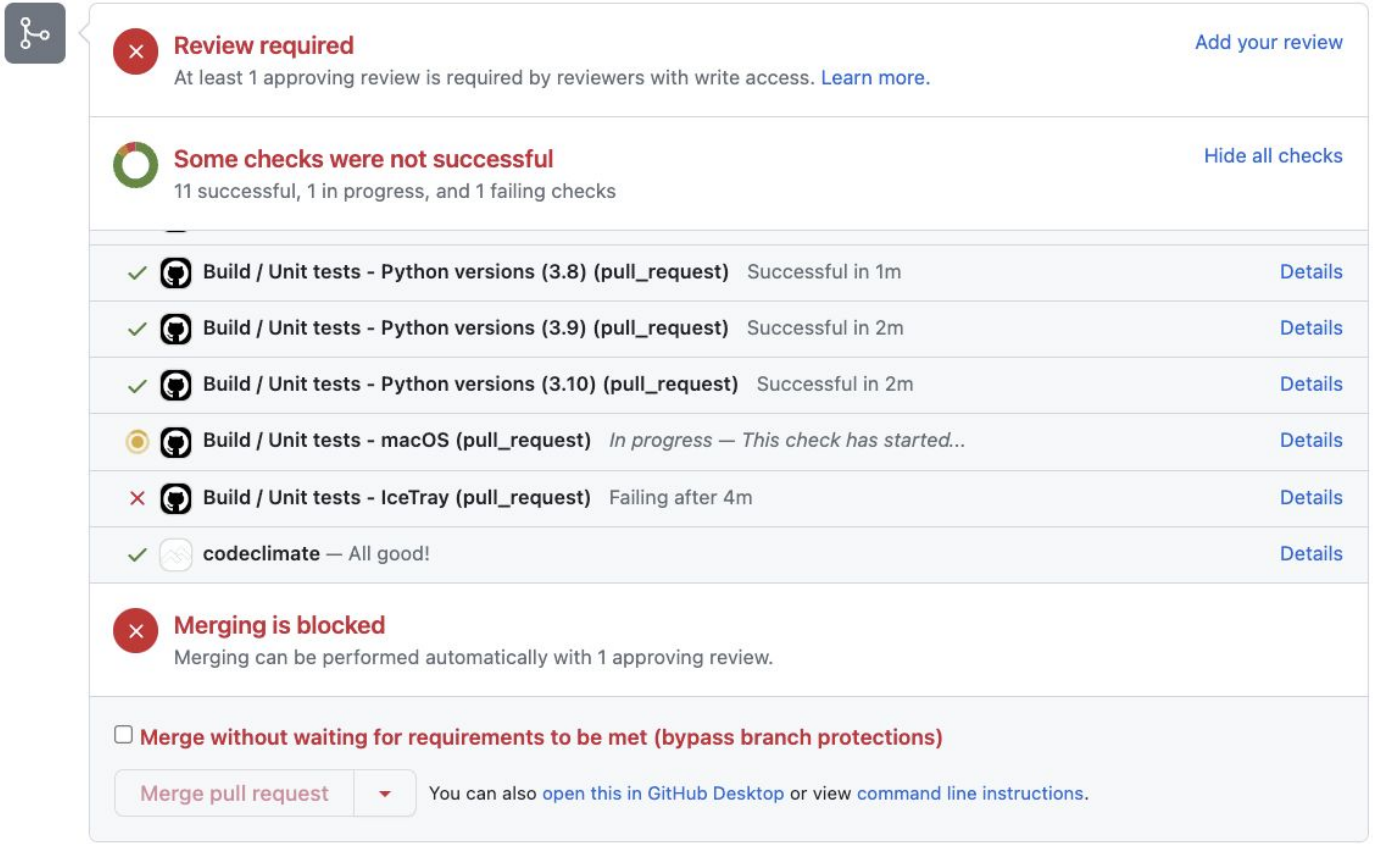
Usage	Development
slack	Build passing
DOI 10.5281/zenodo.6720188	Code quality passing
License Apache 2.0	code style black
python 3.7 | 3.8 | 3.9 | 3.10	maintainability B
version v0.2.3	test coverage 56%

About

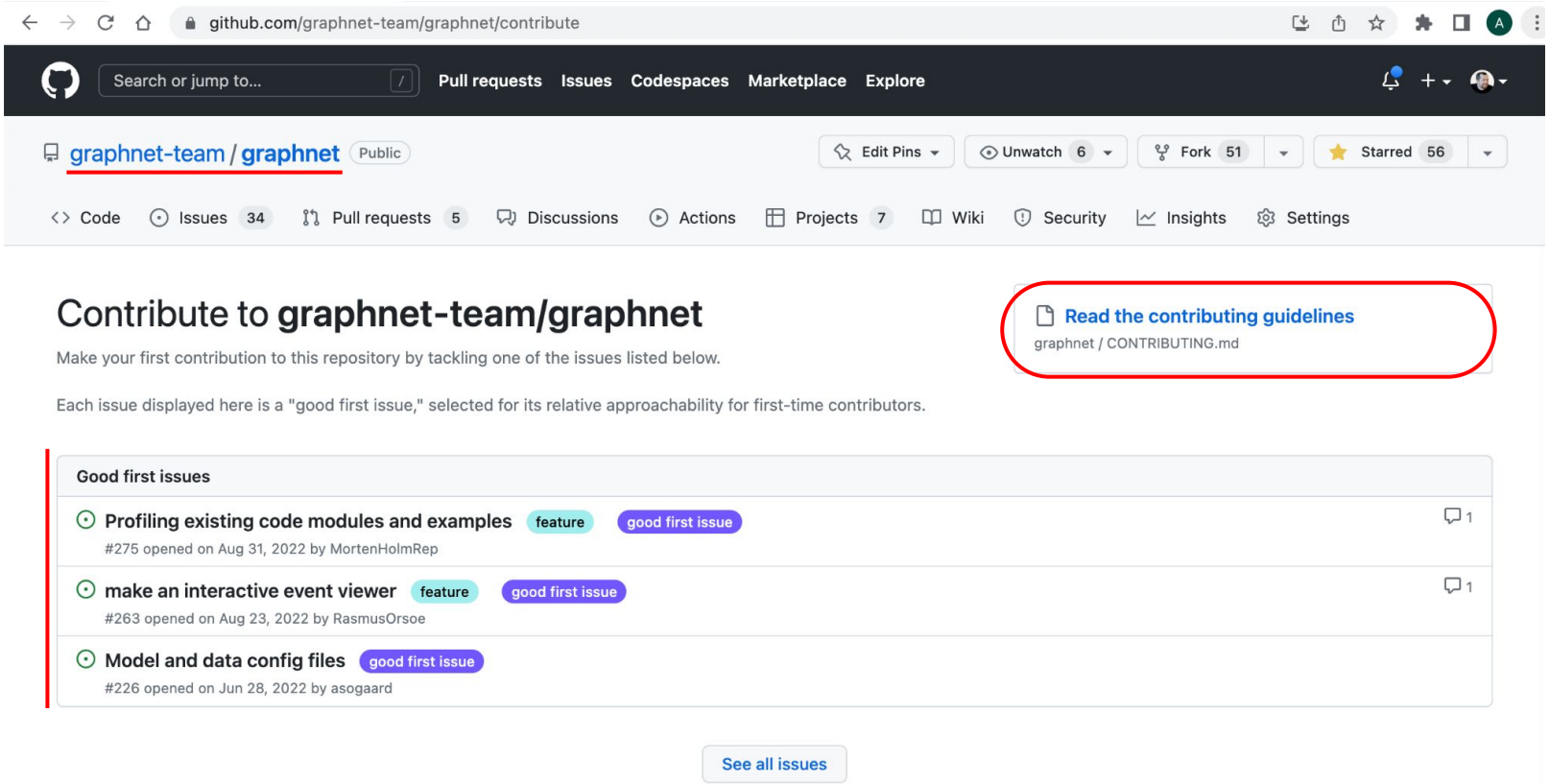
GraphNeT is an open-source Python framework aimed at providing high quality, user friendly, end-to-end functionality to perform reconstruction tasks at neutrino telescopes using graph neural networks (GNNs). GraphNeT makes it fast and easy to train complex models that can provide event reconstruction with state-of-the-art performance, for arbitrary detector configurations, with inference times that are orders of magnitude faster than traditional reconstruction techniques.


Pulse
Contributors
Community
Community Standards
Traffic
Commits
Code frequency
Dependency graph
Network
Forks
People
Forks
graphnet-team / graphnet
akkarimi / graphnet
AMHermansen / graphnet
Andreas-MJ / graphnet
ArturoLlorente / graphnet
Aske-Rosted / graphnet
asogaard / graphnet
gtc1030 / graphnet
atamazian / graphnet
BozianuLeon / graphnet
Camhock101 / graphnet
cddeangelis / graphnet
dollabillgates / graphnet
edenni / graphnet
FNoaGut / graphnet
grburgess / graphnet
gurusarath1 / graphnet
HiroakiHibi / graphnet
HiroakiHibi-ARISE / graphnet
Hivecluster901 / graphnet
I8dNLo / graphnet
ilymak / graphnet
jlkjlkj / graphnet
JostMigenda / graphnet
jprottung / graphnet
justinvasel / graphnet
kaareendrup / graphnet
kayla-leonard / graphnet
KhalidAl-otaibi / graphnet
About
Graph neural networks for neutrino telescope event reconstruction
graphnet-team.github.io/graphnet/
python
machine-learning
deep-learning
neural-network
gpu
pytorch
physics-analysis
astrophysics
high-energy-physics
particle-physics
neutrino-oscillations
neutrinos
graph-neural-network
neutrino-physics
Readme
Apache-2.0 license
Cite this repository
56 stars
6 watching
50 forks
Report repository
Releases
7 tags
Create a new release
Packages 1
graphnet
Contributors 13
+ 2 contributors
Environments 1
github-pages
Active

Code quality



How to start contributing



First public release (v1.0)

Prepared for submission to JINST

Graph Neural Networks for Low-Energy Event Classification & Reconstruction in IceCube

R. Abbasi,[16] M. Ackermann,[62] J. Adams,[17] N. Aggarwal,[24] J. A. Aguilar,[11] M. Ahlers,[21] M. Ahrens,[52] J.M. Alameddine,[22] A. A. Alves Jr.,[30] N. M. Amin,[42] K. Andeen,[40] T. Anderson,[58,59] G. Anton,[25] C. Argüelles,[13] Y. Ashida,[38] S. Athanasiadou,[62] S. Axani,[14] X. Bai,[48] A. Balagopal V.,[38] M. Baricevic,[38] S. W. Barwick,[29] V. Basu,[38] R. Bay,[7] J. J. Beatty,[19,20] K.-H. Becker,[61] J. Becker Tjus,[10] J. Beise,[60] C. Bellenghi,[26] S. Benda,[38] S. BenZvi,[50] D. Berley,[18] E. Bernardini,[46] D. Z. Besson,[33] G. Binder,[7,8] D. Bindig,[61] E. Blaufuss,[18] S. Blot,[62] F. Bontempo,[30] J. Y. Book,[13] J. Borowka,[0] C. Boscolo Meneguolo,[46] S. Böser,[39] O. Botner,[60] J. Böttcher,[0] E. Bourbeau,[21] J. Braun,[38] B. Brinson,[5] J. Brostean-Kaiser,[62] R. T. Burley,[1] R. S. Busse,[41] M. A. Campana,[47] E. G. Carnie-Bronca,[1] C. Chen,[5] Z. Chen,[53] D. Chirkin,[38] K. Choi,[54] B. A. Clark,[23] L. Classen,[41] A. Coleman,[42] G. H. Collin,[14] A. Connolly,[19,20] J. M. Conrad,[14] P. Coppin,[12] P. Correa,[12] S. Countryman,[44] D. F. Cowen,[58,59] R. Cross,[50] C. Dappen,[0] P. Dave,[5] C. De Clercq,[12] J. J. DeLaunay,[57] D. Delgado López,[13] H. Dembinski,[42] K. Deoskar,[52] A. Desai,[38] P. Desiati,[38] K. D. de Vries,[12] G. de Wasseige,[35] T. DeYoung,[23] A. Diaz,[14] J. C. Díaz-Vélez,[38] M. Dittmer,[41] H. Dujmovic,[30] M. A. DuVernois,[38] T. Ehrhardt,[39] P. Eller,[26] R. Engel,[30,31] H. Erpenbeck,[0] J. Evans,[18] P. A. Evenson,[42] K. L. Fan,[18] A. R. Fazely,[6] A. Fedynitch,[56] N. Feigl,[9] S. Fiedlschuster,[25] A. T. Fienberg,[59] C. Finley,[52] L. Fischer,[62] D. Fox,[58] A. Franckowiak,[10] E. Friedman,[18] A. Fritz,[39] P. Fürst,[0] T. K. Gaisser,[42] J. Gallagher,[37] E. Ganster,[0] A. Garcia,[13] S. Garrappa,[62] L. Gerhardt,[8] A. Ghadimi,[57] C. Glaser,[60] T. Glauch,[26] T. Glüsenkamp,[25] N. Goehlke,[31] J. G. Gonzalez,[42] S. Goswami,[57] D. Grant,[23] S. J. Gray,[18] T. Grégoire,[59] S. Griswold,[50] C. Günther,[0] P. Gutjahr,[22] C. Haack,[26] A. Hallgren,[60] R. Halliday,[23] L. Halve,[0] F. Halzen,[38] H. Hamdaoui,[53] M. Ha Minh,[26] K. Hanson,[38] J. Hardin,[14,38] A. A. Harnisch,[23] P. Hatch,[32] A. Haungs,[30] K. Helbing,[61] J. Hellrung,[0] F. Henningsen,[26] L. Heuermann,[0] S. Hickford,[61] C. Hill,[15] G. C. Hill,[1] K. D. Hoffman,[18] K. Hoshina,[38,a] W. Hou,[30] T. Huber,[30] K. Hultqvist,[52] M. Hünnefeld,[22] R. Hussain,[38] K. Hymon,[22] S. In,[54] N. Iovine,[11] A.

[2209.03042]

The Journal of Open Source Software

GraphNeT: Graph neural networks for neutrino telescope event reconstruction

Andreas Søgaard [1¶], **Rasmus F. Ørsøe** [2], **Leon Bozianu**[1], **Morten Holm**[1], **Kaare Endrup Iversen**[1], **Tim Guggenmos**[2], **Martin Ha Minh** [2], **Philipp Eller** [2], **and Troels C. Petersen** [1]

1 Niels Bohr Institute, University of Copenhagen, Denmark **2** Technical University of Munich, Germany
¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Dan Foreman-Mackey

Reviewers:

- @JostMigenda
- @GageDeZoort

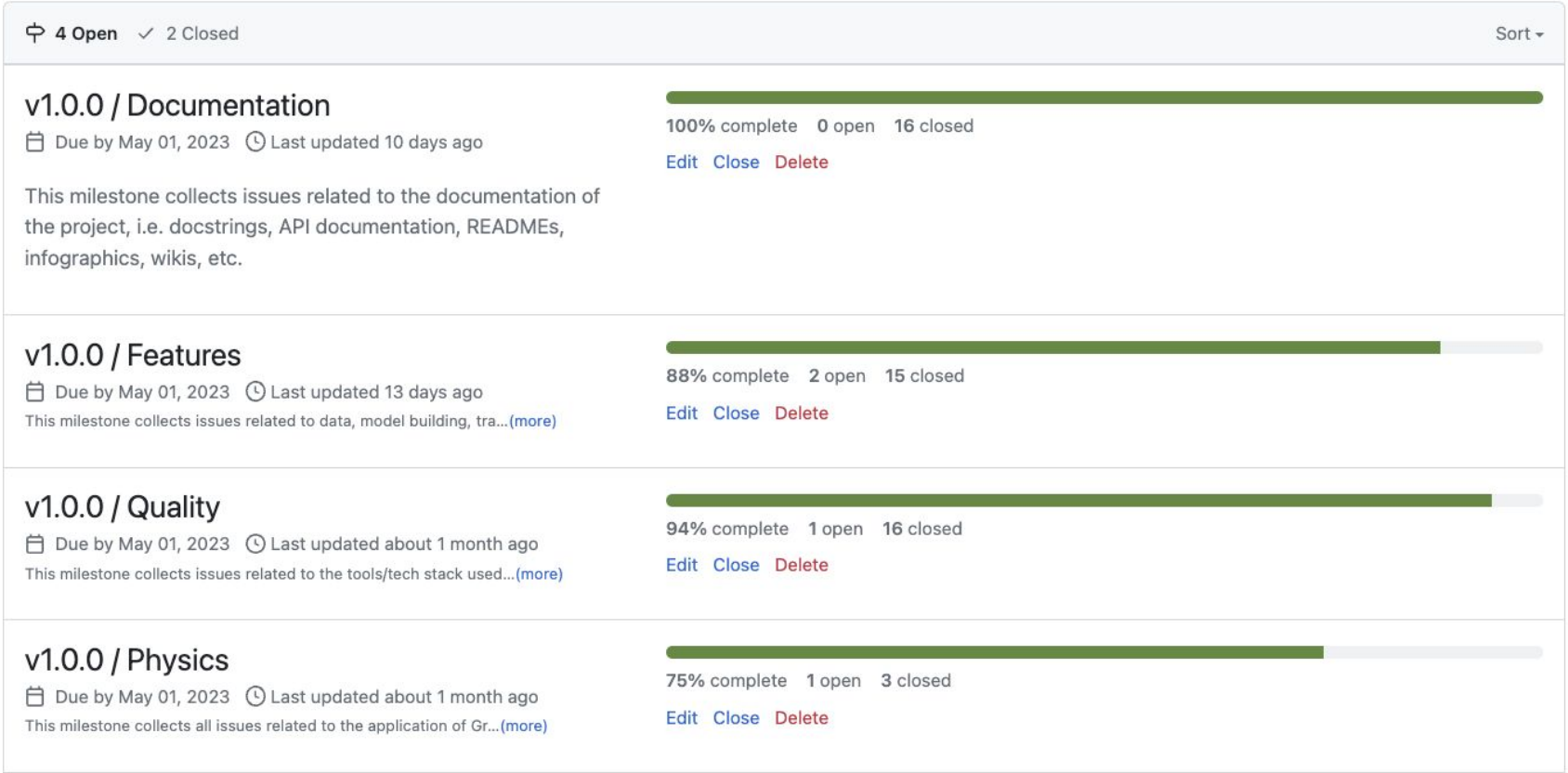
Submitted: 06 October 2022
Published: unpublished

Summary

Neutrino telescopes, such as ANTARES (ANTARES Collaboration, 2011b), IceCube (IceCube Collaboration, 2012, 2017), KM3NeT (KM3NeT Collaboration, 2016), and Baikal-GVD (Baikal-GVD Collaboration, 2018) have the science goal of detecting neutrinos and measuring their properties and origins. Reconstruction at these experiments is concerned with classifying the type of event or estimating properties of the interaction.

`GraphNeT` (Søgaard et al., 2022) is an open-source Python framework aimed at providing high quality, user friendly, end-to-end functionality to perform reconstruction tasks at neutrino telescopes using graph neural networks (GNNs). `GraphNeT` makes it fast and easy to train complex models that can provide event reconstruction with state-of-the-art performance, for arbitrary detector configurations, with inference times that are orders of magnitude faster than traditional reconstruction techniques (IceCube Collaboration, 2022a).

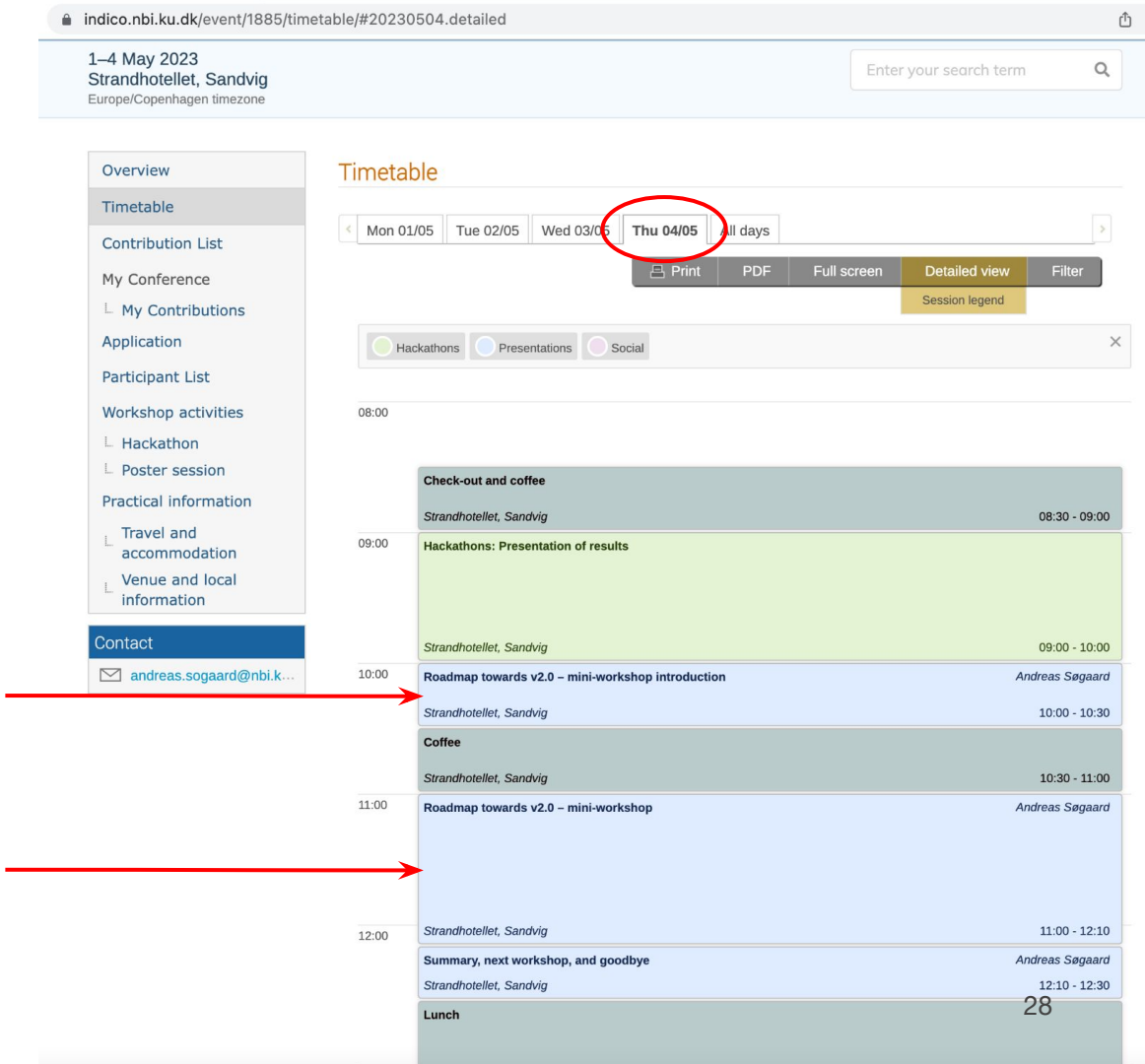
GNNs from `GraphNeT` are flexible enough to be applied to data from all neutrino telescopes, including future projects such as IceCube extensions (IceCube-Gen2 Collaboration, 2017, 2021; IceCube-PINGU Collaboration, 2014) or P-ONE (P-ONE Collaboration, 2020). This means that GNN-based reconstruction can be used to provide state-of-the-art performance on most reconstruction tasks in neutrino telescopes, at real-time event rates, across experiments and physics analyses, with vast potential impact for neutrino and astro-particle physics.



We aim to release GraphNeT v1.0 during this week

Roadmap to v2.0

You can help decide!



GraphNeT

Graph Neural Networks for
Neutrino Telescope Event Reconstruction

Usage	Development
slack	Build passing
DOI 10.5281/zenodo.6720188	Code quality passing
License Apache 2.0	code style black
python 3.7 | 3.8 | 3.9 | 3.10	maintainability B
version v0.2.3	test coverage 56%