

F_TORCH: ENABLING ONLINE TRAINING FOR LARGE-SCALE FORTRAN MODELS

Niccolò Zanotti^{1,2}, Joe Wallwork¹, Jack Atkinson¹, Tom Meltzer¹, Dominic Orchard^{1,3}

¹Institute of Computing for Climate Science, University of Cambridge, UK

²University of Bologna, Italy

³University of Kent, UK

Overview


Scientific models in the climate domain are predominantly developed in Fortran for its high-performance and science-oriented nature, while much of Machine Learning (ML) research is conducted in Python owing to its rich ecosystem of ML frameworks, most notably PyTorch.

The scientific community is increasingly moving towards a hybrid modeling approach replacing traditional physical parametrizations of sub-grid scale processes with ML-based alternatives that use neural networks trained on observational or simulation data, reducing the computational cost and potentially improving accuracy.

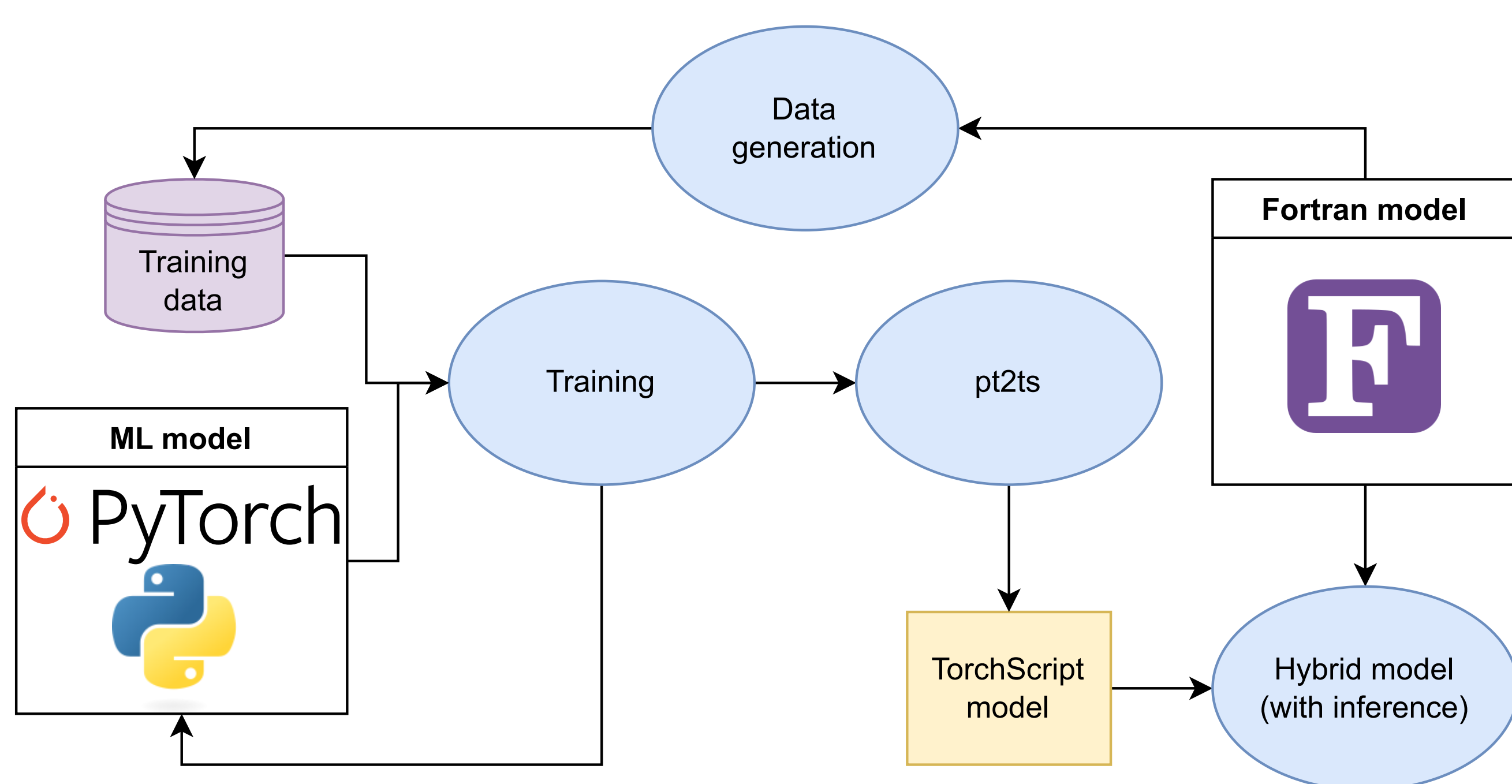
This shift towards a data-driven approach poses engineering challenges as models are developed by different groups in different programming languages.


F_Torch (Atkinson et al., 2025) is an open source library developed by *ICCS* that enables hybrid modeling by seamlessly interfacing PyTorch ML models with Fortran code.

Features

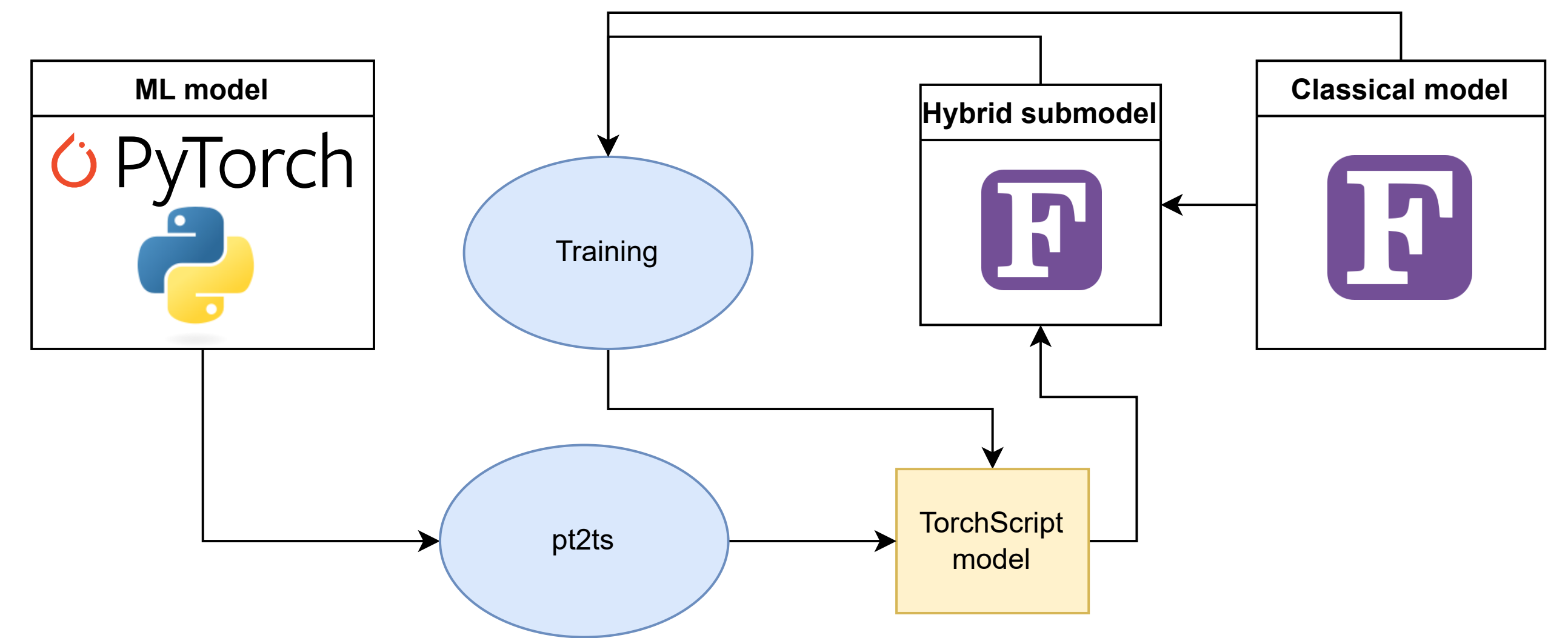
- Binds Fortran to PyTorch's C++ backend (no  runtime) via `iso_c_binding` module
- Provides a user-friendly Fortran API close to PyTorch API
- Enables *zero-copy* data transfer between languages
- Handles Fortran *column-major* vs. C/C++ *row-major* memory layout differences
- Supports CPU and GPU (CUDA, XPU, HIP, MPS backends)
- Supports model serialization
- Tested on UNIX and Windows operating systems

Hybrid Modeling Offline



- **Design the ML model in PyTorch** with PyTorch  API
- **Dataset generation** for the ML model training via Fortran-based model runs
- **Train** the PyTorch model on generated data, optimizing for the specific predictive tasks
- **pt2ts** Convert trained models (`.pt`) to TorchScript (`.ts`)
- **Hybrid model** Integrate the TorchScript model with Fortran code using F_Torch, enabling inference during runs

Proposed Online Workflow



Benefits:

- No -Fortran conversion overhead + direct integration

Current state:

- Tensor operator overloading, `requires_grad` support
- `torch_tensor_backward` (backpropagation) with gradient extraction and zeroing

In progress:

- Expose `torch::optim` algorithms and loss functions
- Expose components from `torch::nn::Module`

Case studies using F_Torch

- **MiMA-ML**: Found that online behavior diverges from offline for nearly-identical offline gravity waves models. Mansfield and Sheshadri (2024)
- **ICON-ML**: ML convection parameterizations work best when causal relations are eliminated. Heuer et al. (2024)
- **GloSea6**: Replaced BiCGStab solver with ML to speed up forecasting. Park and Chung (2025)

Ongoing Work and References

- Currently working on coupling a *snow on sea ice emulator* of SnowModel with Icepack (S. Driscoll, J. Atkinson; HPC grant)



- Docs: cambridge-iccs.github.io/F_Torch
- GitHub: [Cambridge-ICCS/F_Torch](https://github.com/Cambridge-ICCS/F_Torch)

References

- Atkinson, Jack et al. (2025). "F_Torch: A Library for Coupling PyTorch Models to Fortran". In: *Journal of Open Source Software* 10.107, p. 7602. DOI: [10.21105/joss.07602](https://doi.org/10.21105/joss.07602).
- Heuer, Helge et al. (Aug. 1, 2024). "Interpretable Multiscale Machine Learning-Based Parameterizations of Convection for ICON". In: *Journal of Advances in Modeling Earth Systems* 16.8, e2024MS004398. DOI: [10.1029/2024MS004398](https://doi.org/10.1029/2024MS004398).
- Mansfield, Laura and Aditi Sheshadri (2024). "Uncertainty Quantification of a Machine Learning Subgrid-Scale Parameterization for Atmospheric Gravity Waves". In: *Journal of Advances in Modeling Earth Systems* 16.7. DOI: [10.1029/2024MS004292](https://doi.org/10.1029/2024MS004292).
- Park, Hyesung and Sungwook Chung (2025). "Utilization of a Lightweight 3D U-Net Model for Reducing Execution Time of Numerical Weather Prediction Models". In: *Atmosphere* 16.1. DOI: [10.3390/atmos16010060](https://doi.org/10.3390/atmos16010060).

Contact: niccolo.zanotti@studio.unibo.it | [website](https://www.niccolozanotti.com)

Acknowledgements

This project is supported by Schmidt Sciences, LLC.
This project received funding from C2D3-Accelerate programme.