Some General Takeaways from the DUNE Core Software & Computing + AI/ML Joint Workshop

A Partial Summary

FNAL NPC Journal Club
April 5th, 2026

by J. L. Barrow
with key inputs from Callum Wilkinson (LBNL)

Thanks to Aaron, Ilker and Rice for hosting us!

And to everyone for the work and erudite discussions!

Callum Wilkinson, a_sort_of_summary.pdf

WARNING

I am a *user*…

Beyond a few old ~bespoke grid campaigns & bit of ML,
I am in no way a computing expert…

Computing needs are a HIGHLY COUPLED problem

?How to distinguish between?
user analysis + group analysis + reconstruction development
individual production + group production
individual ML train/inference + reco chain train/inference

I hope this talk serves to *inform* as best as possible, and connect experts with group leaders, lead analyzers, and other users

Apologies if things get a bit repetitive at times across this and other talks

Talk Timetable

Topics	Speakers
DUNE Physics Working Group Needs	C. Marshall, J. Barrow, J. Calcutt
ProtoDUNE+FD+ND Simulation & Reconstruction	L. Paulucci, M. Kramer, J. Calcutt
Detector Simulation & Reconstruction Software (WC, Pandora, SPINE)	H. Yu, L. Whitehead, F. Drielsma
Some ML Applications (NuGraph, Transformers)	G. Cerati, J. Liu, J. Barrow
Current ML Libs., Future PHLEX Libs., GPU Resources Available	V Hewes, A. Bashyal, K. Herner
Workflows, Production, and Data Management; FAIR Principles	A. McNab, A. Higuera, S. Timm
Foundation Models, LLMs, Inference as a Service	K. Terao, A. Rafique, D. Sagar, M. Bhattacharya
ML Trigger, Signal Processing, Agents	C. Hasnip, A. Bashyal, A. Higuera
GPU-based Optical Simulations, Differentiable Programming	I. Parmaksiz, K. Terao
Genesis Mission	L. Whitehead, J. Bian
Remarks	C. Wilkinson

Central Purposes

- Understand current scope of planned analysis work within WGs
  - Resources expected: storage, CPU hours
- Understand *potential* scope of future work
  - How many GPUs do we need if we get all that we want?
  - How built-in to our reconstruction frameworks do GPUs need to be?

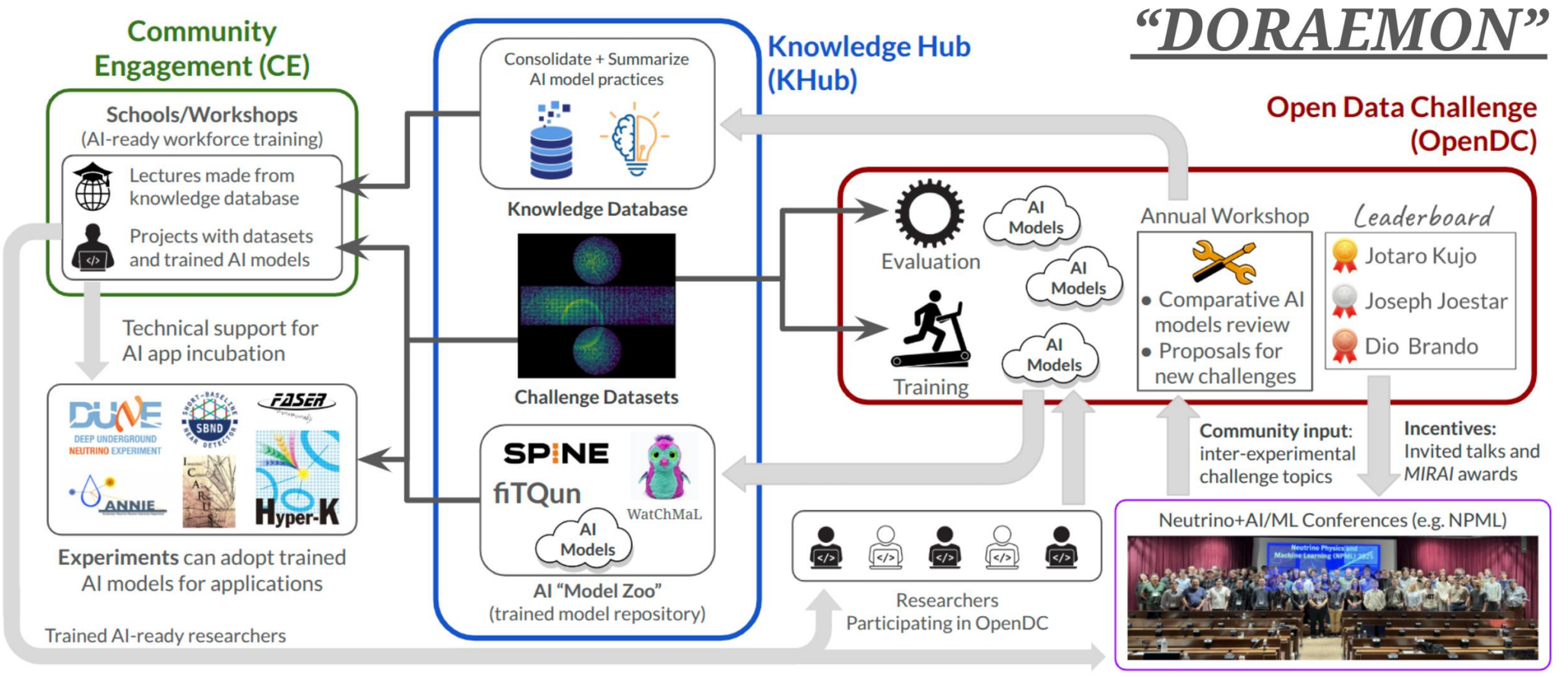
- Reviews of some ongoing cutting-edge ML work across DUNE
  - Largely can’t show this here, sorry!

- Reviews of best practices we should endeavor to utilize as a community!

- GENESIS Mission…

Needs for research ecosystem for a common FM

- Diverse datasets, tasks, and common benchmarks
- Synthesize fundamental knowledge about FM-for-neutrino/HEP datasets
- Public datasets and trained models made available for applications R&D



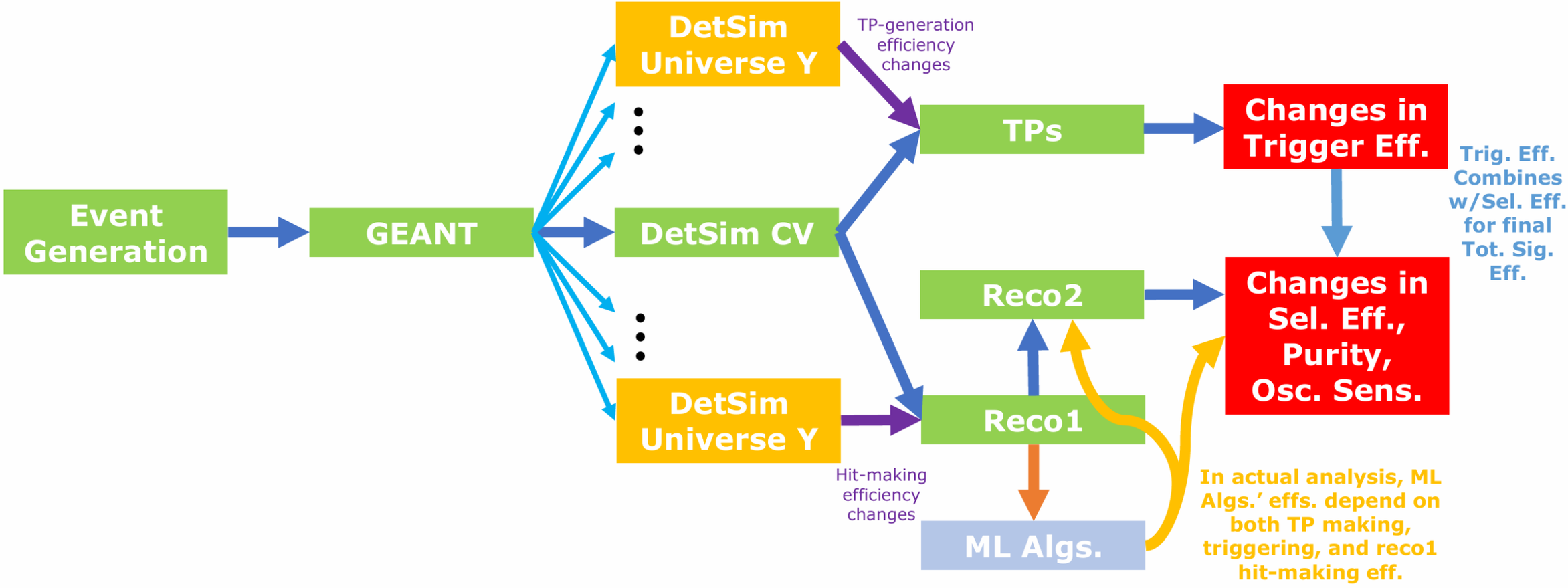
Kazuhiro Terao,
https://docs.google.com/presentation/d/1Hiwdxu92oyR-z4TBalDL-Ri6okCQQ_tM2VGs1IjYmEo/edit?usp=sharing

Our findings from group members

- Never a bad time to buy nvidia shares
- GPU usage generally is going to go up
- GPUs are being used for acceleration, as well as ML
- A100 is the most popular hardware, but this is driven by availability on EAF
- Available FNAL resources becoming increasingly maxed out
  - EAF GPU slots are increasingly becoming fully occupied

Laura Paulluci, DUNEAIMLWorkshop_fdsimreco.pdf

Detector Systematics' Effects on Simulation & Reconstruction Chain



Proposal: for each DetSim "universe", run standard reco chain along with TP generation

GPUs, NERSC and other extra computing needs

- GPU usage generally increasing rapidly in DUNE
- FD is often a testbed for AI developers to test ideas
- DUNE FD computing have been modest in terms of processing power
  - Most event topologies are ‘simple’ to simulate, but we need a lot of statistics
  - So far have not needed HPC centres
- Educated guess is that this will start to change in the coming years
  - Processing low energy + radiologicals in the full 10kt sim may need HPC
  - Some collaborators expressed desire to use GPU during sim stage (e.g. Opticks/Celeritas)
  - GPU usage will generally go up

Laura Paulluci, DUNEAIMLWorkshop_fdsimreco.pdf

Expected types of GPU usage at DUNE FD

We are starting to hear about two kinds of GPU usage at DUNE FD, with very different needs

1. AI/ML usage to enhance sim/reco performance
   - Requires infrequent but intense GPU usage outside of production to train ML networks
   - Typically does not require GPUs for inference -> only need CPUs at production time
     - This is not a hard rule
   - Mostly requires interactive-level access to GPUs rather than GPUs on worker nodes

2. GPU-accelerated algorithms to enhance processing speed
   - Run heavy algorithms on a GPU that are CPU infeasible
   - Requires interactive-level GPUs (for development) and GPUs on worker nodes (for production)

Laura Paulluci, DUNEAIMLWorkshop_fdsimreco.pdf

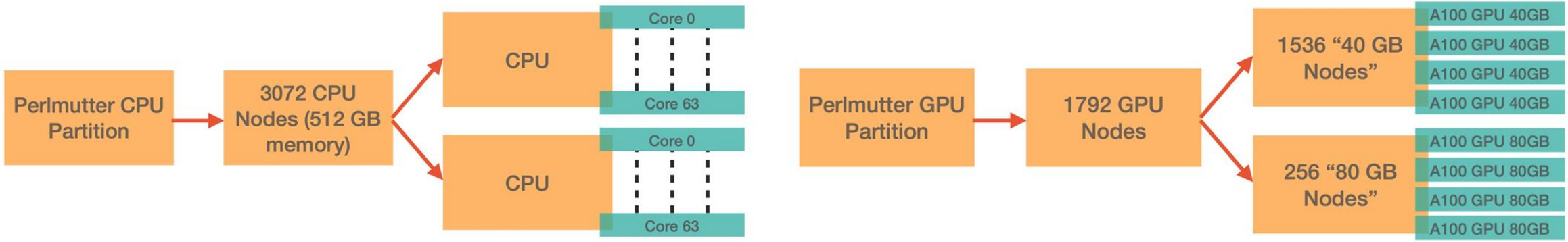
Primary contact: A. Higuera

Scintillation light full optical simulation

- Full optical simulations are prohibitively slow on CPUs
  - Fast optical methods (semi-analytic, optical libraries, GNN) are implemented to save time and resources
- Explorations in DUNE to use Opticks and Celeritas, to GPU-accelerate the full optical simulation
- Current development efforts use local RTX GPUs (5090 and 4090), and DUNE's allocation at NERSC used to test portability
- GPU's would be needed for FD simulation production that make use of this method
  - Current timeline for development completion is summer 2026
  - Will rely on justIN to deploy workflows that should provide DUNE-GPUs from OSG and NERSC allocation

Laura Paulluci, DUNEAIMLWorkshop_fdsimreco.pdf

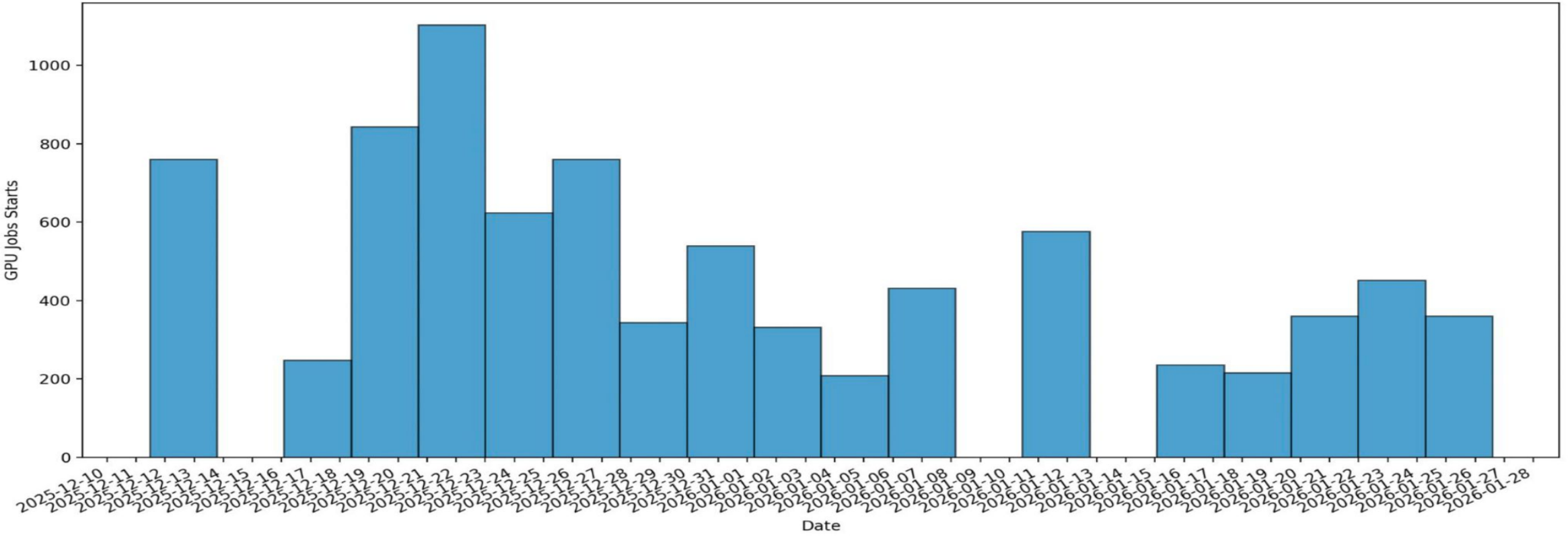
Perlmutter @ NERSC



- Abundant source of both CPU and GPU nodes, initially introduced as a solution to file transfer bottleneck.

Matt Kramer, 20260310_RiceWorkshop_GPUsAtND

Perlmutter @ NERSC



- Abundant source of both CPU and GPU nodes, initially introduced as a solution to file transfer bottleneck.
- Make use of the **DUNE Production** (m3249) allocation.
  - Used 20k GPU hours in the last two months of 2025, made possible by hours top-up in second week of December (**unable to spend it all**).

Matt Kramer, 20260310_RiceWorkshop_GPUsAtND

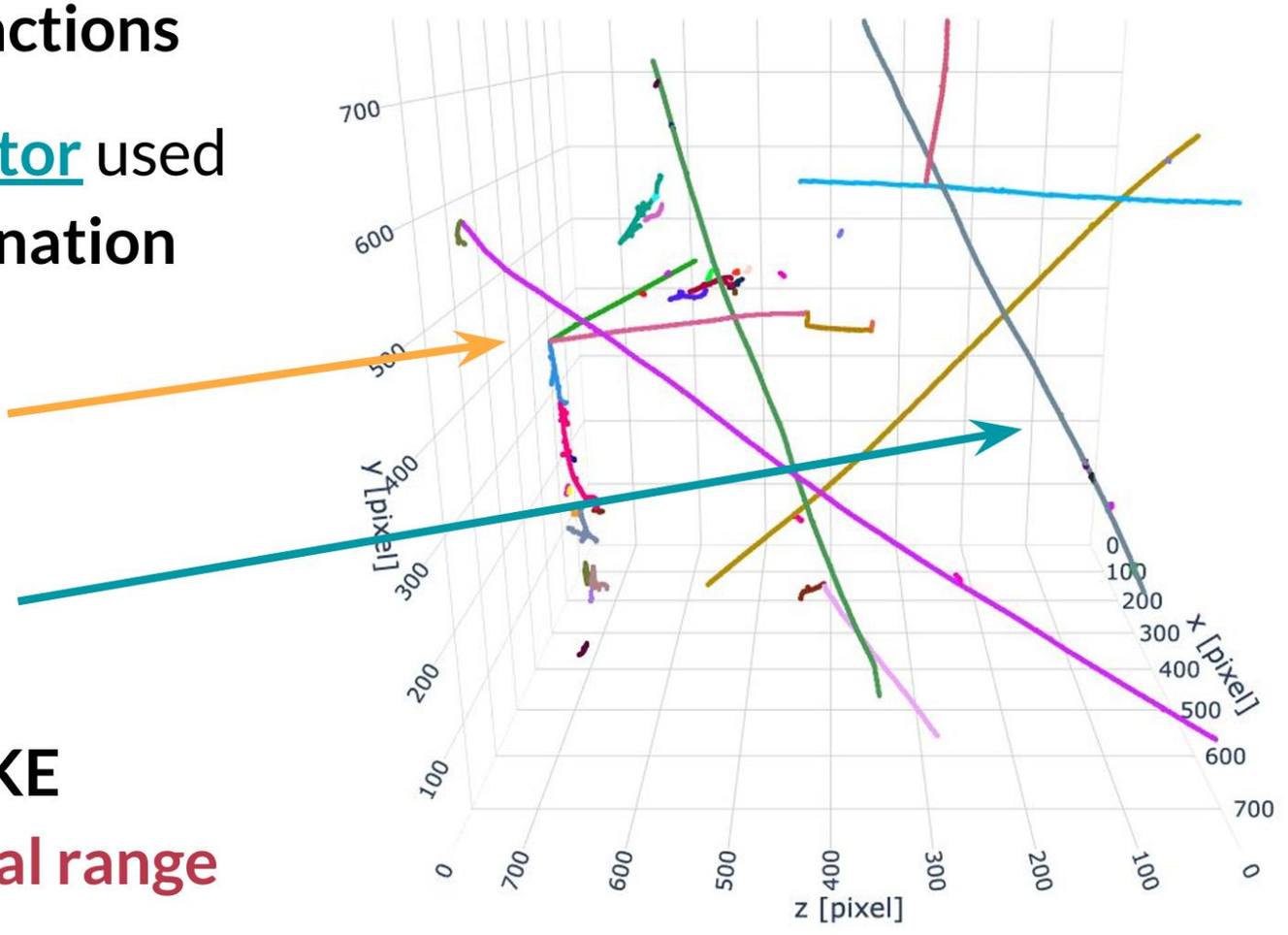
Training Sets

SPINE is a **supervised model** crucially
***NOT* trained on neutrino interactions**

It is trained using the **DLPGenerator** used in **each detector simulation declination**

- Multi-Particle Vertex (MPV)
  = Particle bombs
- Multi-Particle Rain (MPR)
  = Single particle gun

Isotropic angle, multiplicity and KE sampled uniformly within physical range



Portability

SPINE **dependencies packaged** in an Apptainer/Docker image here

SPINE is tagged and released on PyPI (pypi.org/project/spine)

Each training process yields **one inference config + one weight file**

- **Inference configuration** shared through spine-prod (tagged)
  - Ships with tagged version of SPINE for reproducibility
  - Ships with trivial submit.py for end users (slurm-only for now)
- **Weight files (~110 MB per file)** hosted publicly
  - Tagged, download + chesum for validation by spine-prod

Documentation on ReadTheDocs: spine.readthedocs.io

https://hub.docker.com/r/deeplearnphysics/larcv2
http://pypi.org/project/spine/
https://github.com/DeepLearnPhysics/spine-prod
http://submit.py/
https://s3df.slac.stanford.edu/data/neutrino/spine/weights/
http://spine.readthedocs.io/

GPUs on HTC Clusters

- Several sites have GPUs available; accessible via usual job submission commands for IF/CF expts (separate for CMS)
  - Simply add --lines='+RequestGPUs=1' to your Jobsub submission.
  - **As of now, user is responsible for specifying appropriate container** (use the --singularity-image option and make sure the image is available to the worker, usually via CVMFS).
    - If using justIN, your job script will need to invoke any additional containers manually (or bring along any needed GPU libs in a tarball)
  - See Andrew's talk for the details of doing the equivalent in JustIN
- HTCondor classads are possible to match if you need a specific type of GPU.
  - FermiGrid and several DUNE UK sites have A100s; some on OSG opportunistically
    - **I had successful test jobs at QMUL and Manchester this week**
- Several advantages to using these resources
  - **Very little competition for them** (i.e. no long queues a la NERSC)
  - No finite allocations!
- Have to copy your outputs back like any other grid job

NERSC and HEPCloud

- Perlmutter (both CPU and GPU queues) available via HEPCloud for currently onboarded experiments (includes DUNE)
  - **Contact the production group for running your campaigns**
- Rucio storage now working there as well
- Triton Inference server setups available
- No support for MPI workflows currently within HEPCloud (but we have very few of these right now); they require direct slurm submit, but we can do that too.
- If you have a larger workflow, Production can probably run it for you!

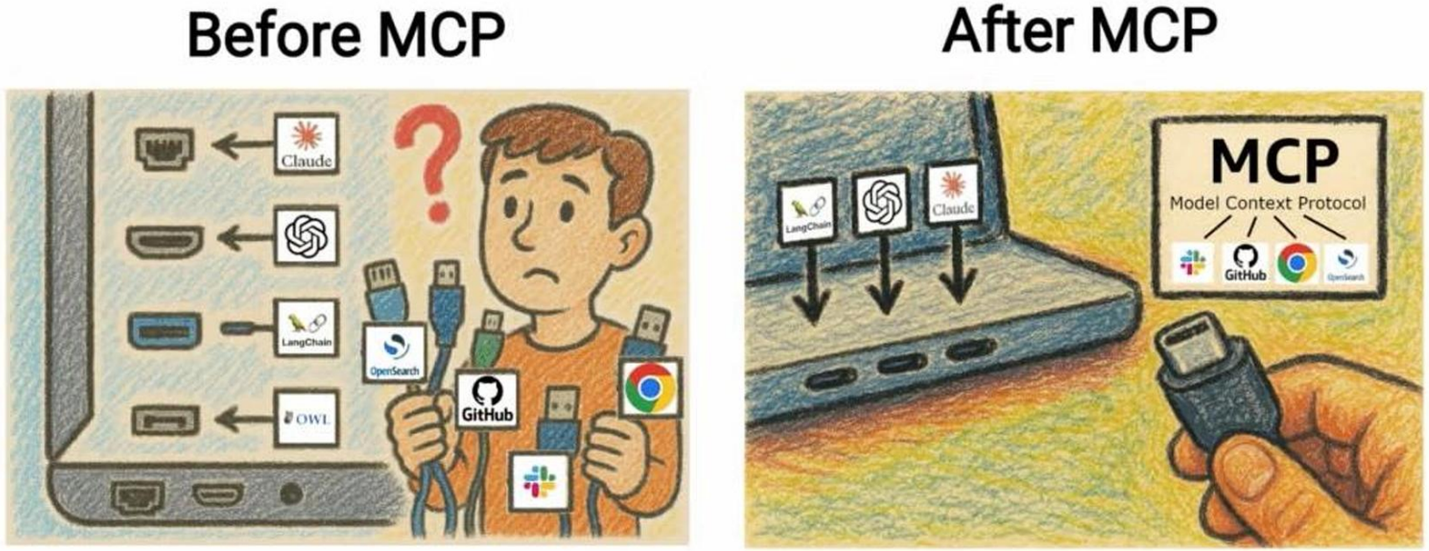
Genesis Mission / American Science Cloud

- DUNE members involved both in the science demo side and the Infrastructure Provider side
- Several DUNE collaborators involved in a demo scheduled for May (Mike Wang, Meghna Bhattacharya, others?)
- “ Inference as a Service for HEP and NP Data Analysis ”
  - Includes a run of “nugraph” on open-release MicroBooNE data
  - Larsoft-based SN reconstruction workflow with an ML-based data reduction step at the beginning.
- Staff from the Fermi Data Platform (including S. Timm) involved in demoing access to the data
- (much more details on the physics of the demo in Meghna’s talk coming later).
- In general a lot of emphasis on the business of tokenization for AI/ML like structures

Steven Timm, 2026Mar11_aiml_talk.pdf

AI/ML in Data management ops

- Recently, we explored a possible approach to using MCP/LLMs in Rucio and OpenSearch, could be a potential AI/ML application in Data Management Ops
- MCP (Model Context Protocol) is a standardized communication protocol connecting AI agents (LLMs) with external tools and data sources, AI can interact with multiple tools without writing custom code for each.
- MCP Approach
  - Unified interface layer with a standard JSON specification
  - Flow: Host (AI Apps) ⇄ MCP Client ⇄ MCP Server (Data Source)
  - Secure: Enables controlled, local-first access to sensitive data



https://opensearch.org/blog/introducing-mcp-in-opensearch/

Steven Timm, 2026Mar11_aiml_talk.pdf

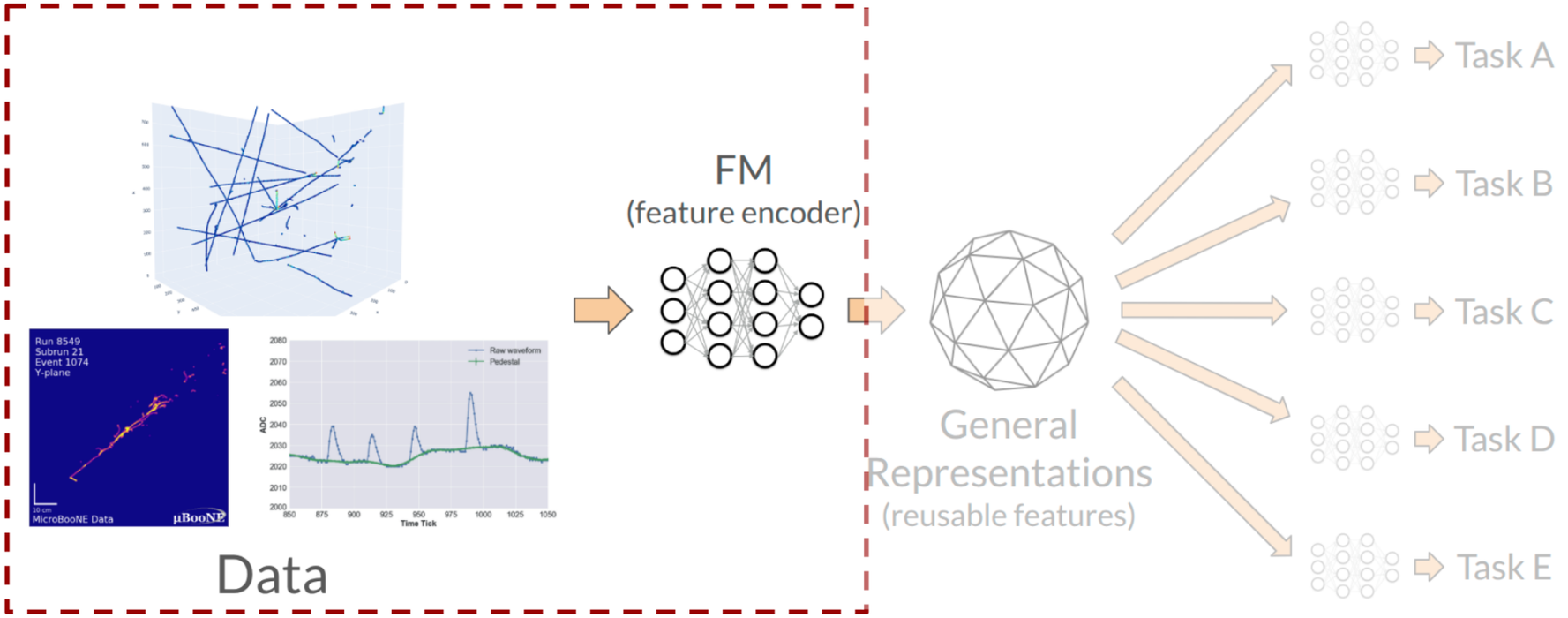
What do we expect for AI/ML

- More smaller files?
- More instances of many processes reading the same file?
- High I/O workflows? (Num cpu instructions / bytes of IO) < 100?
- Perversely random reads?
- Lots of writing of small files?

What is the possible unexpected?

- The questions for us:
- Is our current data model enough to sustain increasing MC and data processing for new AI/ML developments?
- How can data management help a FAIR distribution of training data and AI/ML models within the collaboration?

FM: Fine-Tuning Paradigm

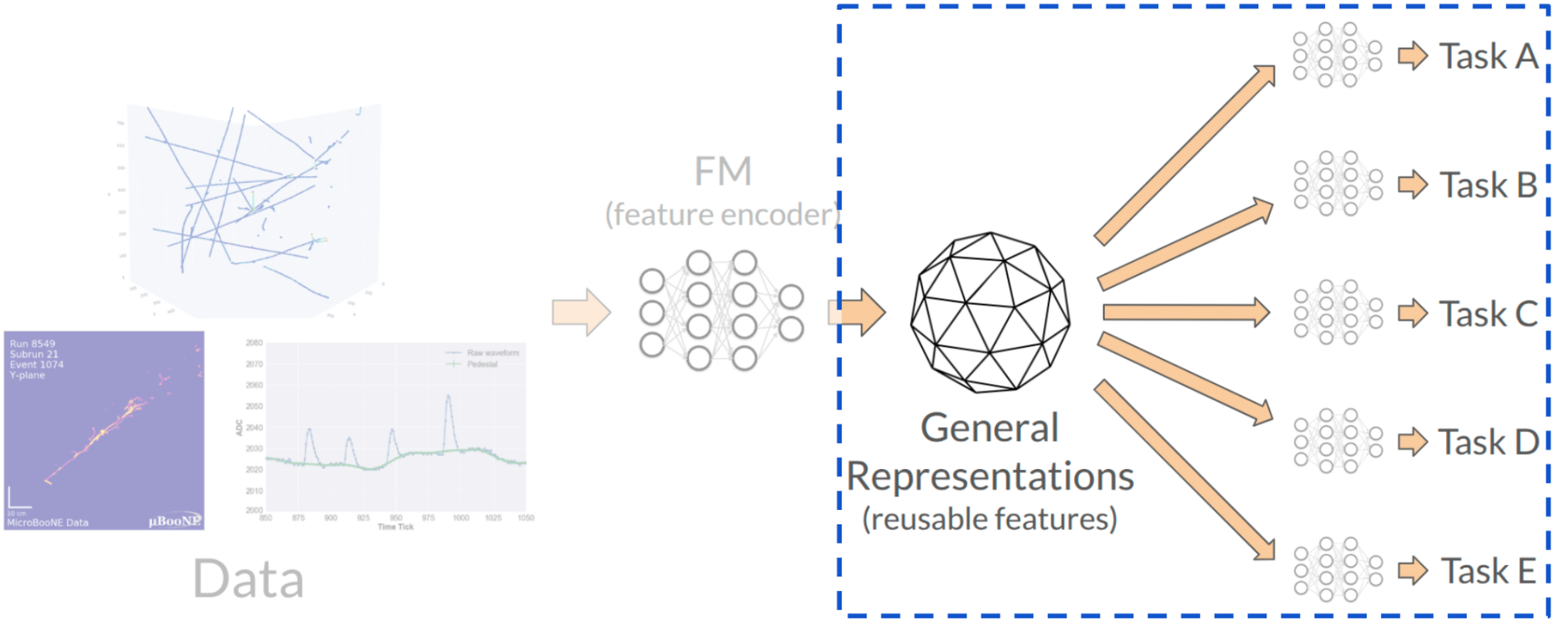


Pre-training = big model/compute, shared engineering effort

- Collaboration-wide
- Inter-experimental model (e.g. across DUNE & SBN)
- Generic detector physics (e.g. LHC/DUNE)

Kazuhiro Terao,
https://docs.google.com/presentation/d/1Hiwdxu92oyR-z4TBalDL-Ri6okCQQ_tM2VGs1IjYmEo/edit?usp=sharing

FM: Fine-Tuning Paradigm

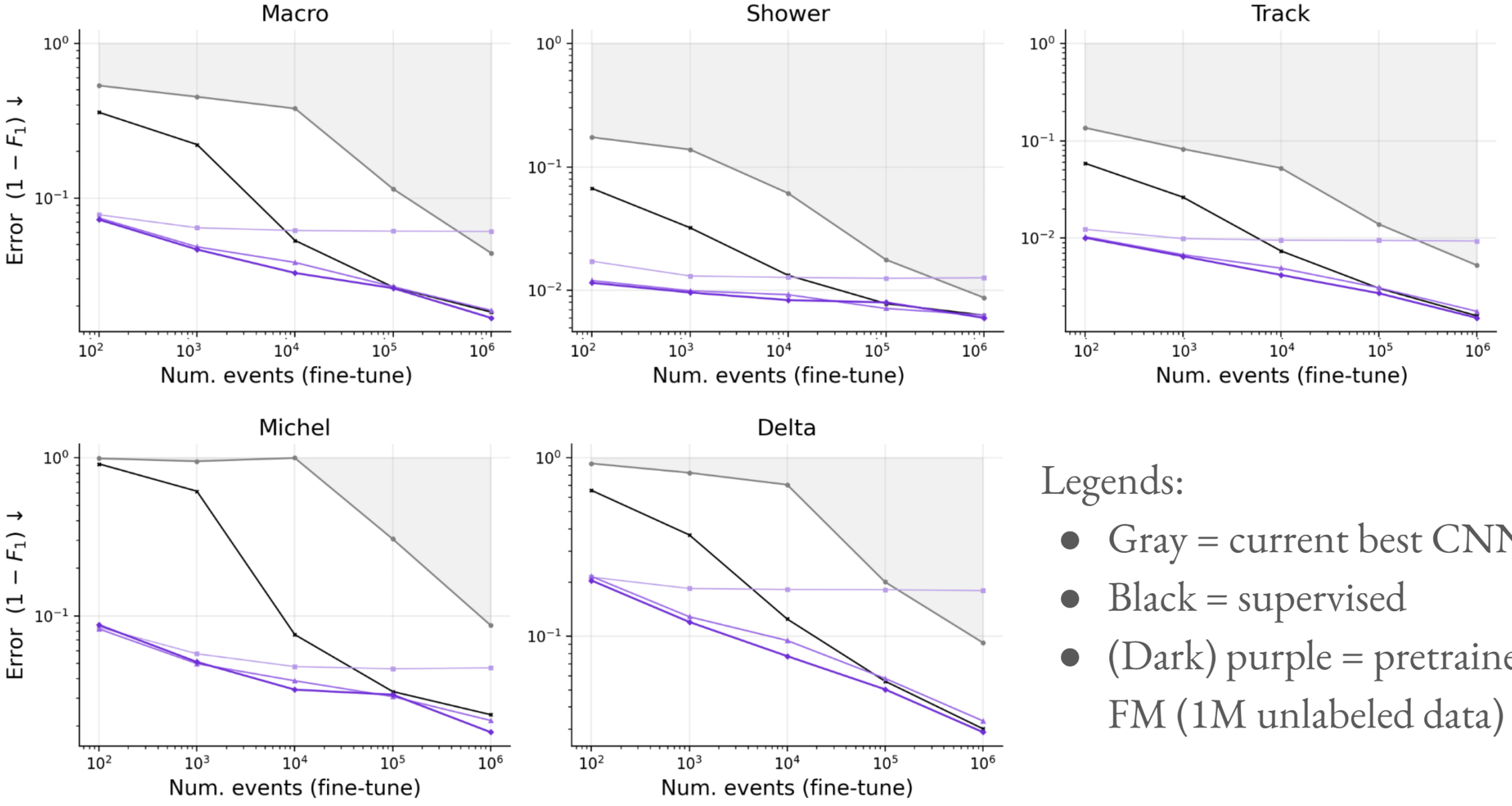


Fine-tuning = small (manageable), task-specific models

- Reconstruction/Analysis as "fine-tuning"
- Expand AI research scope (e.g. couple with LLMs)
  - General representation = "AI-ready data"

Kazuhiro Terao,
https://docs.google.com/presentation/d/1Hiwdxu92oyR-z4TBalDL-Ri6okCQQ_tM2VGs1IjYmEo/edit?usp=sharing

Fine Tuning: Data Scalability + Benchmark v.s. Supervised



Legends:

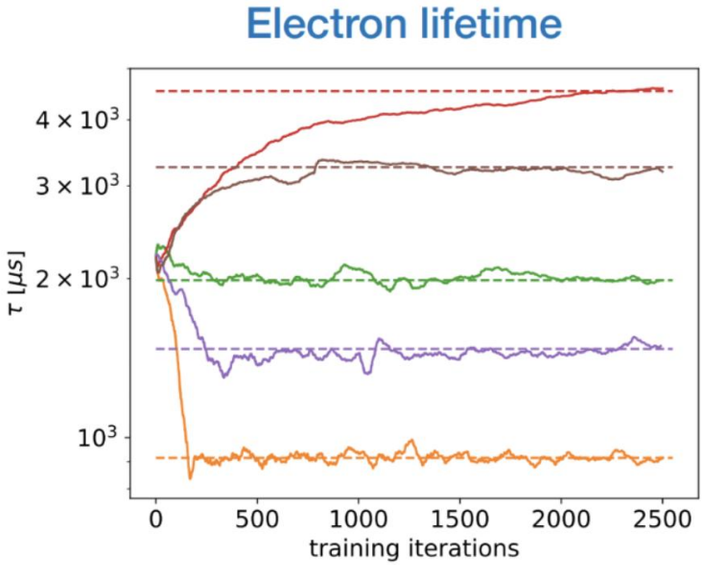
- Gray = current best CNN
- Black = supervised
- (Dark) purple = pretrained FM (1M unlabeled data)

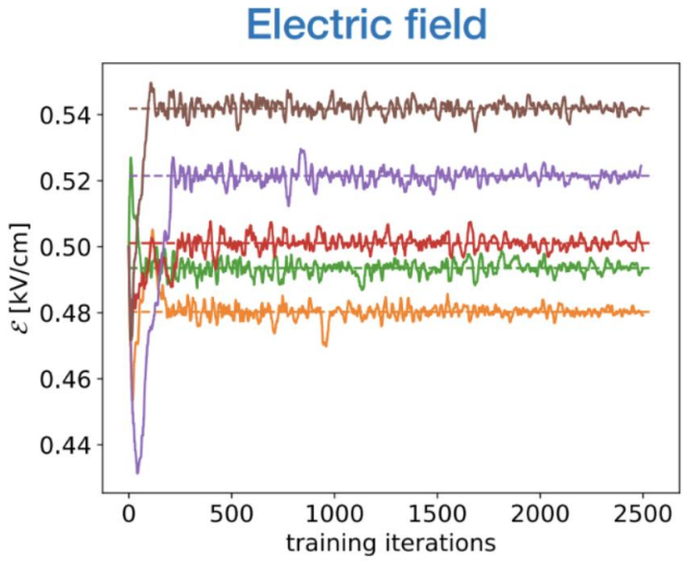
Kazuhiro Terao,
https://docs.google.com/presentation/d/1Hiwdxu92oyR-z4TBalDL-Ri6okCQQ_tM2VGs1IjYmEo/edit?usp=sharing

Differentiable TPC+LArPix Simulator


Yifan Chen


Pierre Granger

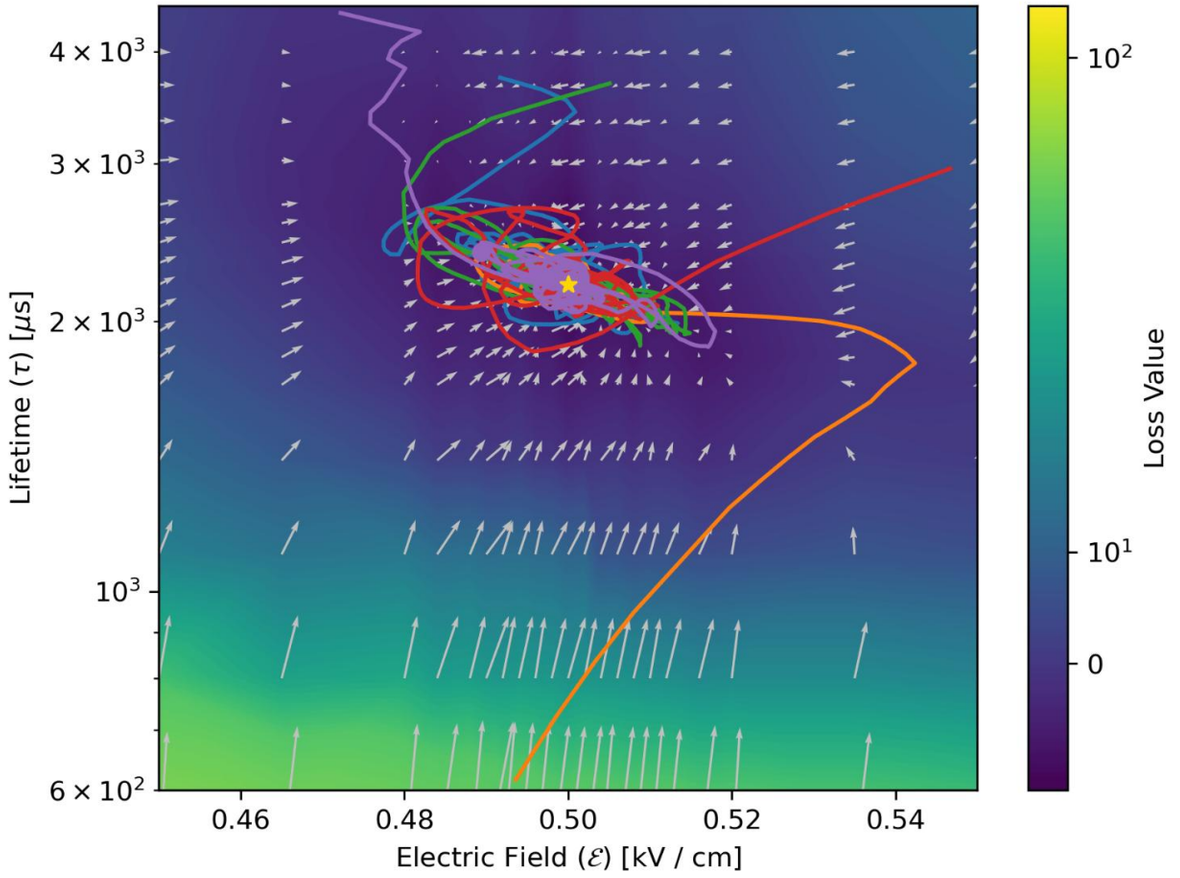




Using muon (MIP) tracks, calibrate two params

Left: same initial value + different target values

Right: different initial values + same target



10

Reference

Kazuhiro Terao, Differentiable Programming

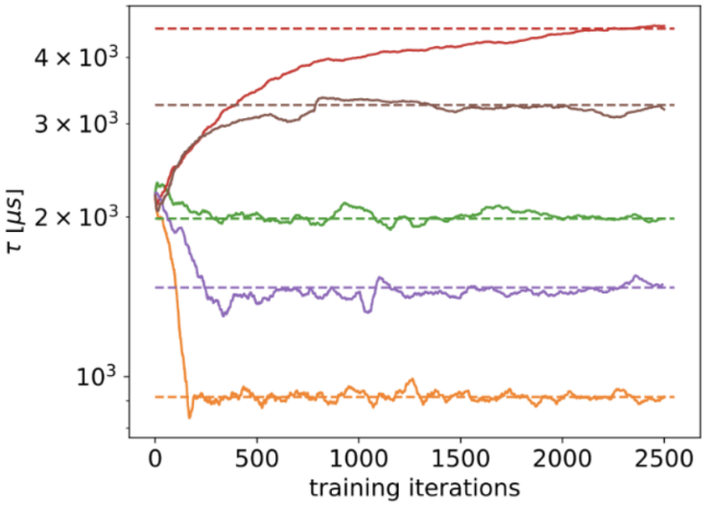
[2309.04639] Differentiable Simulation of a Liquid Argon Time Projection Chamber

Differentiable TPC+LArPix Simulator

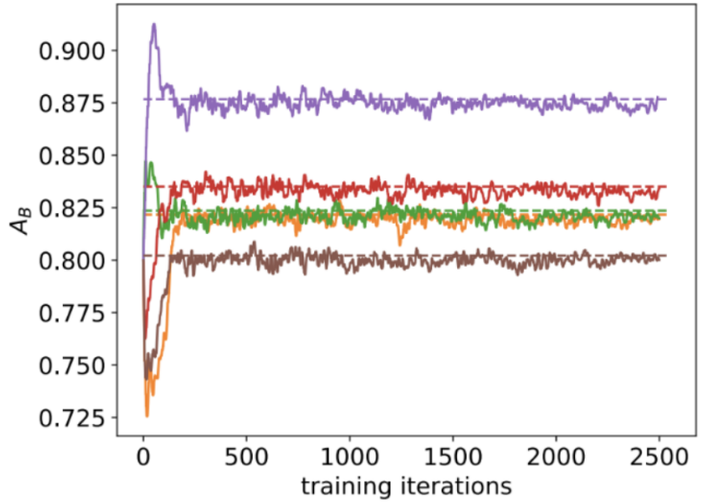

Yifan Chen


Pierre Granger

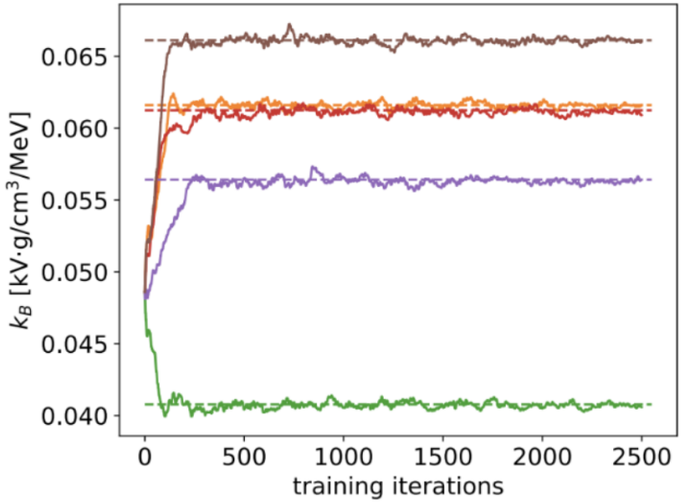
Electron lifetime



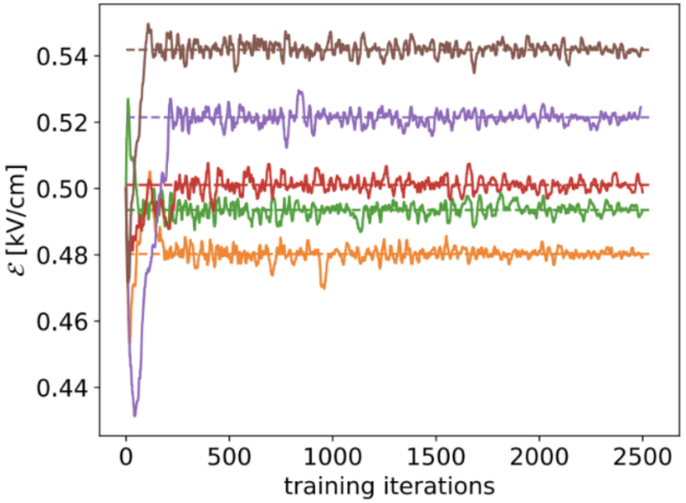
Recombination model A_B



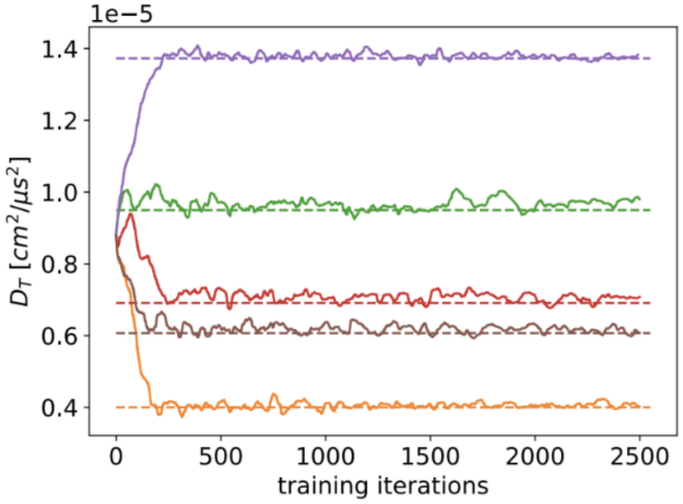
Recombination model k_B



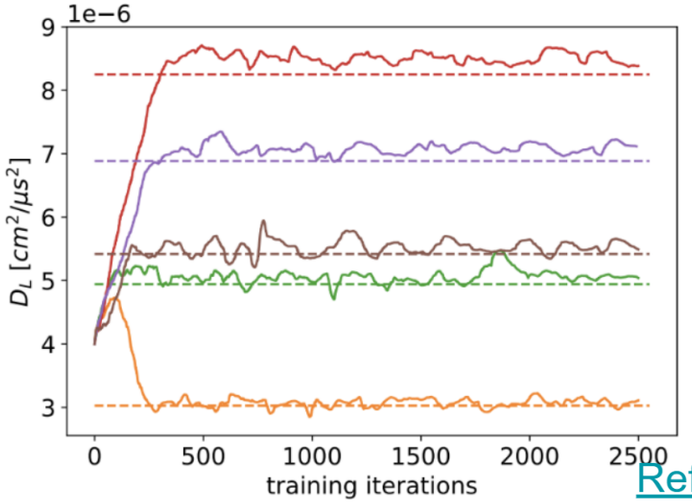
Electric field



Transverse diffusion coefficient



Longitudinal diffusion coefficient



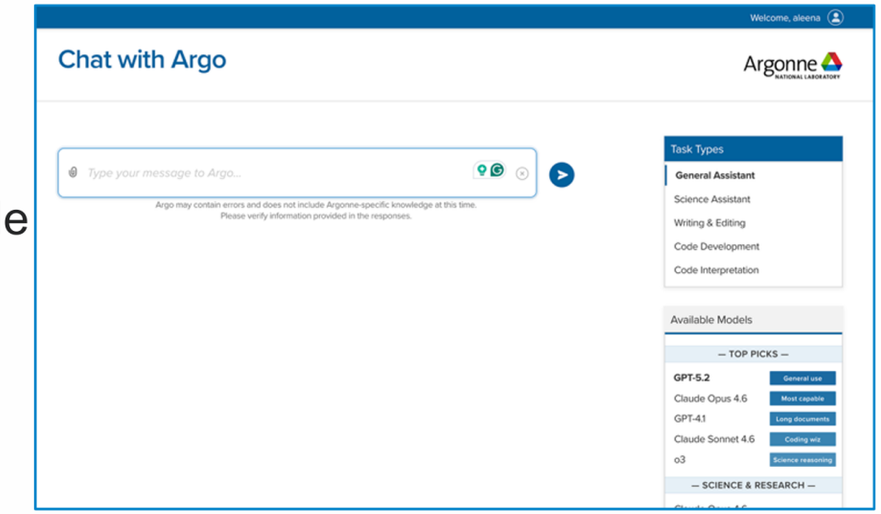
Reference

Kazuhiro Terao, Differentiable Programming

[2309.04639] Differentiable Simulation of a Liquid Argon Time Projection Chamber

LOCAL LLMS AT ANL AND FNAL

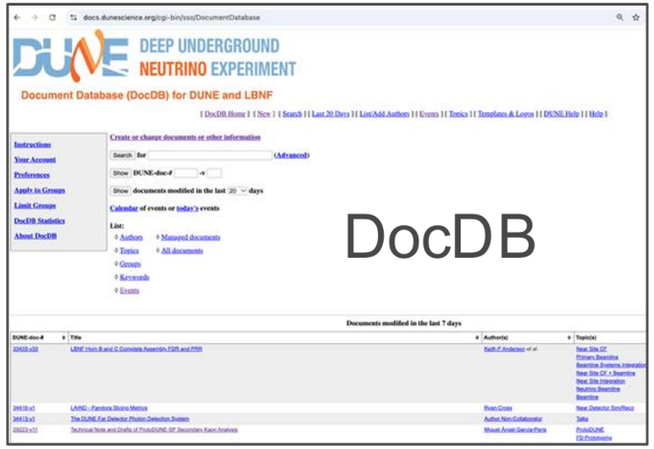
- **Argo at Argonne**
  - Internal **chatbot platform** hosted at Argonne
  - **Security-first**: does not store or share user data
  - **Text-only** capabilities (no image generation yet)
  - **Argo Gateway**: Custom API enabling direct LLM access
  - Supports multiple providers: such as OpenAI, Anthropic, Google
- **vLLM/Ollama at Fermilab**
  - Inference engines that run locally on FNAL hardware
  - vLLM: High-performance **LLM inference server,**
  - Ollama: Easier model management but lower throughput
  - Designed for **LLM development and experimentation**
  - **Data-secure**: no data is shared or stored externally
  - vLLM and Ollama supports models **Qwen, LLaMA, mixtral, and GPT-OSS**

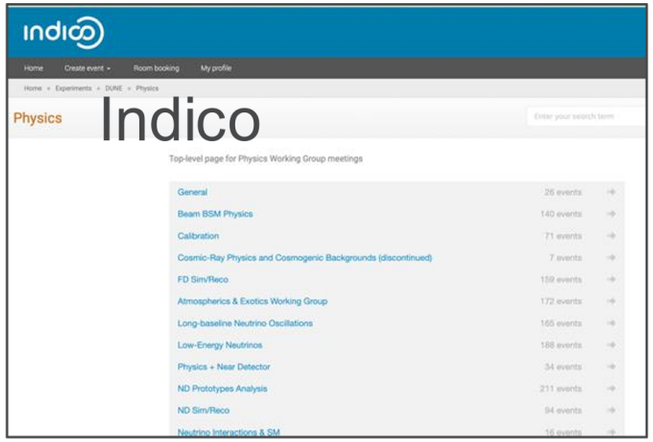


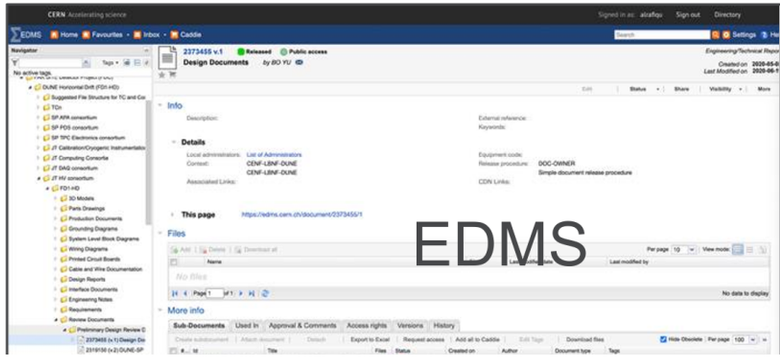
Aleena Rafique, LLM for DUNE Documentation

DUNE DOCUMENTATION

- DUNE relies on multiple documentation platforms, each of them already have tens of thousands of documents:
  - DocDB (Fermilab hosted)
  - EDMS (CERN Hosted)
  - Indico (hosted by various institutions)
- Extracting relevant information across these systems is **non-trivial**
  - Especially Indico, which lacks robust search capabilities
- Developing an LLM (called DUNE-GPT) for DUNE to combine these databases into one, in a form that when a collaborator queries the model, along with responses, it fetches the top matching references from these databases.
- We utilized ANL-local resources/supercomputer (Aurora) to perform initial tasks and for large indexing jobs in the first batch
- This tool can be used for any DUNE-specific information and could be expanded to perform data analyses efficiently in the future

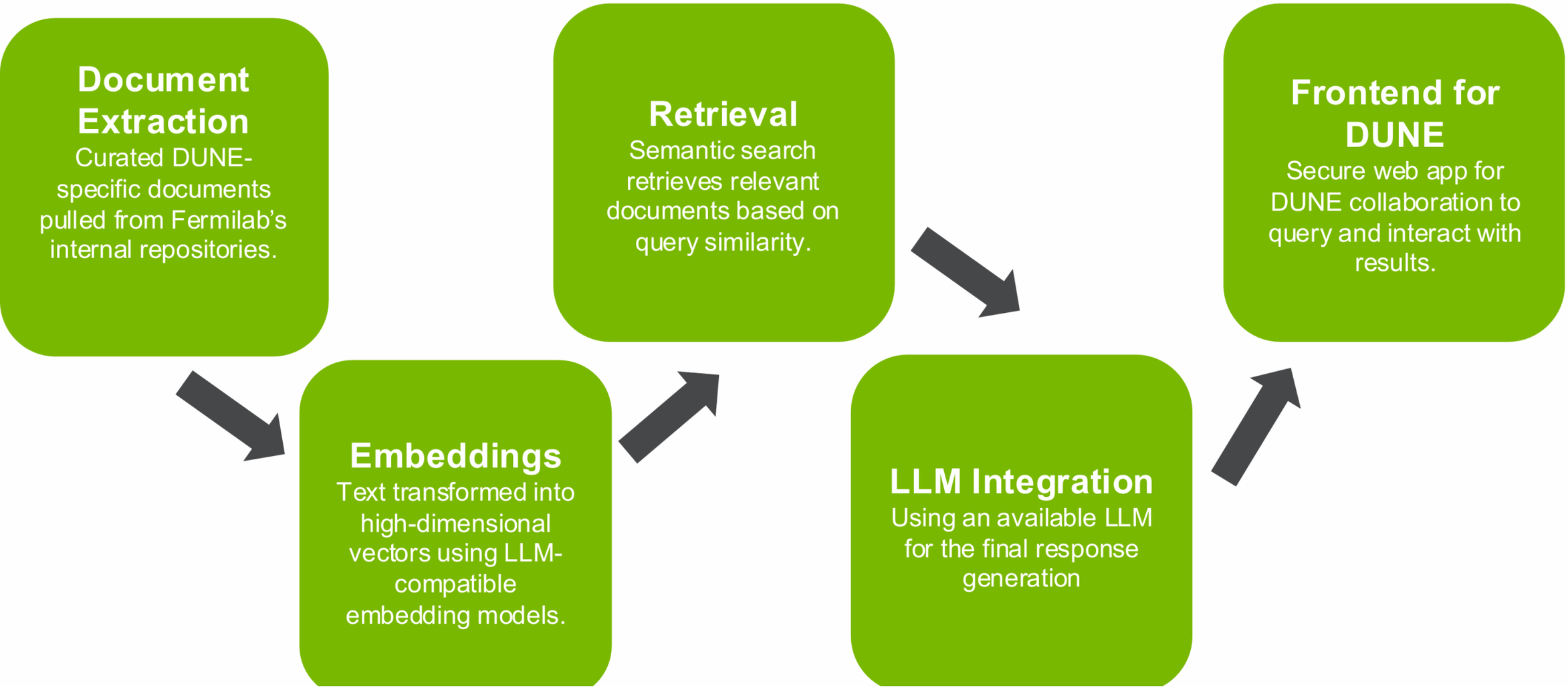






Aleena Rafique, LLM for DUNE Documentation

WORKFLOW



Aleena Rafique, LLM for DUNE Documentation

UTILIZING AURORA FOR LARGE-SCALE EMBEDDING

- The first larger pass of embeddings and indexing leveraged the **Aurora exascale supercomputer** at Argonne National Laboratory (ANL)
  - **5,000 node-hours** allocation under DD award
  - **5 TB** of high-speed storage on the **FLARE** system
- Embedding and indexing jobs were distributed and executed via **Balsam workflow manager**
  - Supports **parallel job execution**, with each job processing separate documents
  - Monitor job progress and resource utilization per node
  - Automatic logging and job tracking for robustness
- First pass from Docdb and Indico documents was processed in **2-3 days** in December 2025.

Aurora System Specifications

Compute Node
2 Intel Xeon CPU Max Series processors: 64GB HBM on each, 512GB DDR5 each; 6 Intel Data Center GPU Max Series, 128GB HBM on each, RAMBO cache on each; Unified Memory Architecture; 8 SlingShot 11 fabric endpoints

CPU-GPU Interconnect
CPU-GPU: PCIe; GPU-GPU: Xe Link

System Performance
Exascale

Platform
HPE Cray EX supercomputer

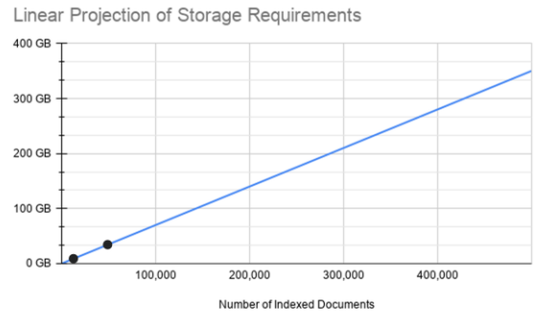
Aleena Rafique, LLM for DUNE Documentation

CURRENT STATUS AND GOALS

- **Currently, DUNE-GPT web interface is:**
  - Hosted locally
    - Testers/developers log into virtual machines with the program installed locally and point queries to localhost
  - Web interface can only be accessed via virtual machines
  - NOT accessible via the internet
  - We are exploring the option of Slack chatbot
- **Goals**
  - Accessible via the internet
  - Collaborators anywhere can access the DUNE-GPT chat interface at a dedicated webpage
    - Actual queries/interaction with the program requires login

WHAT WE NEED

1. Dedicated hosting



- We currently have our code on the dunegpvm
  - The embeddings are stored in */dune/data*
  - It runs fine, a bit slow but okay
- We need a shared repository where more than one person can access the code (maybe with a generic username)
- We anticipate requiring a dedicated host machine moving forward
- Dedicated host machine would need to be:
  - Linux server to host and run Python script
  - Have at least **2-4 CPU cores and 4-8GB RAM**
  - **Large storage** to store a database of training data/reference documents
  - Static external IP address and **exposed to the internet**

Proof of Concept: VLM Fine-Tuned for Neutrino Event Classification and Event Explanation

What We Did

Task: ν_e CC, ν_μ CC or Neutral Current classification from 2-view LArTPC pixel maps (DUNE ND-like geometry)

Model: LLaMA 3.2 Vision 11B fine-tuned with QLoRA

Data: 190,000 simulated events (GENIE + GEANT4); 512×512 grayscale XZ + YZ views

[x,y,z=2x2x7m, beam along +z, ν_μ and ν_e with uniform energy flux 0-10 GeV, pixel-based readout]

Hardware: 4× NVIDIA A6000 (49 GB VRAM each); ~1 week to fine-tune for 1 epoch

Inference: Phrasal constraint beam search → consistent parseable labels + log-prob confidence scores

Key Results

0.87	**0.96**	**29.5M**	**0.85**
Accuracy (vs 0.80 CNN)	AUC-ROC (= ViT-h/14)	Trainable Parameters (QLoRA)	Accuracy 256 x 256 (↓res)

CNN degrades to **0.43 acc** at ↓res | transformers hold at **0.85** → robustness advantage is **real**

Where we are Heading [*Ongoing Work*]

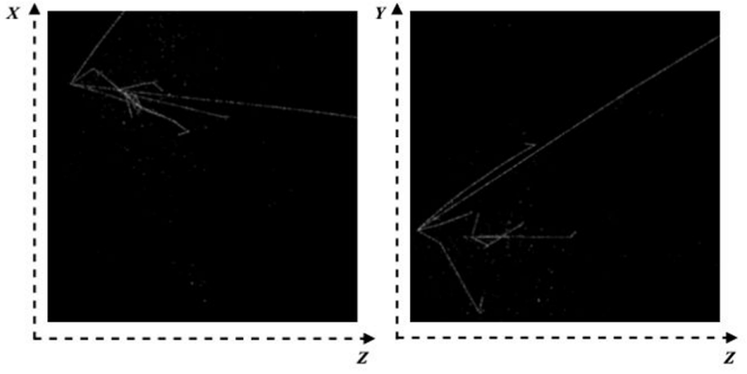
Explicit Explanation Fine-Tuning
Fine-tune LLaMA 3.2 Vision to generate physically grounded, peer-reviewable event descriptions, not just labels.

Pixel Map Segmentation
Single VLM backbone predicts per-pixel particle-type labels (track / shower / vertex / background) simultaneously with classification.

HEP Foundation Model
Single VLM backbone predicts per-pixel particle-type labels (track / shower / vertex / background) simultaneously with classification

Dikshant Sagar, Vision-Language Models for HEP Applications
[2509.08461] Adapting Vision-Language Models for Neutrino Event Classification in High-Energy Physics

True Label: ν_μ CC



Fine-tuned LLaMa 3.2 Vision's Explanation

The pixel maps attached are the z-x and z-y maps of a DUNE Near-Detector-like readouts.
The z-axis is the beam direction. To classify the event as NuE CC, NuMu CC or Neutral Current,
we need to analyze the pixel maps and look for the following features:

For NuE CC, there should be a fuzzy electron shower.
For NuMu CC, the muon track is usually longer and narrow.
For Neutral Current, there is no significant muon track or electron shower.

In the given pixel maps, the muon track is longer and narrow, which suggests that the event is NuMu CC.
Therefore, the classification of the event is NuMu CC.

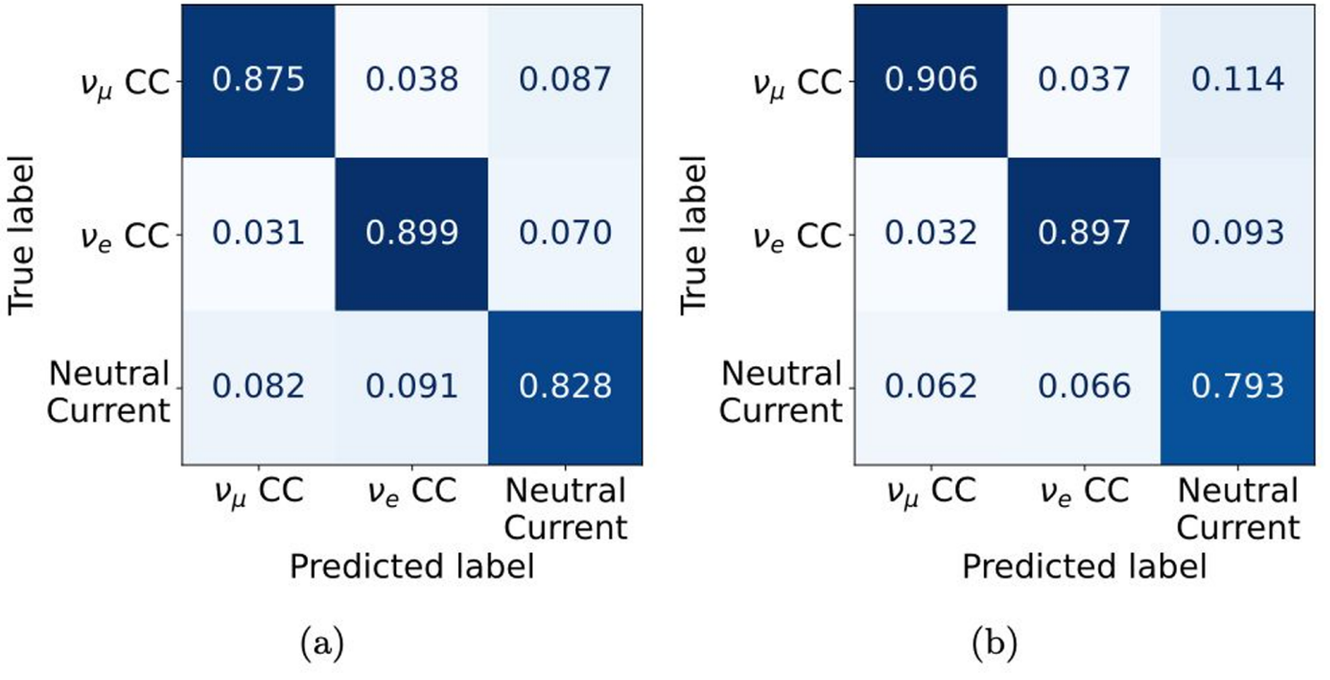


Figure 8: Finetuned LLaMA 3.2 Vision's (a) recall matrix (truth normalized) and (b) precision matrix (prediction normalized).

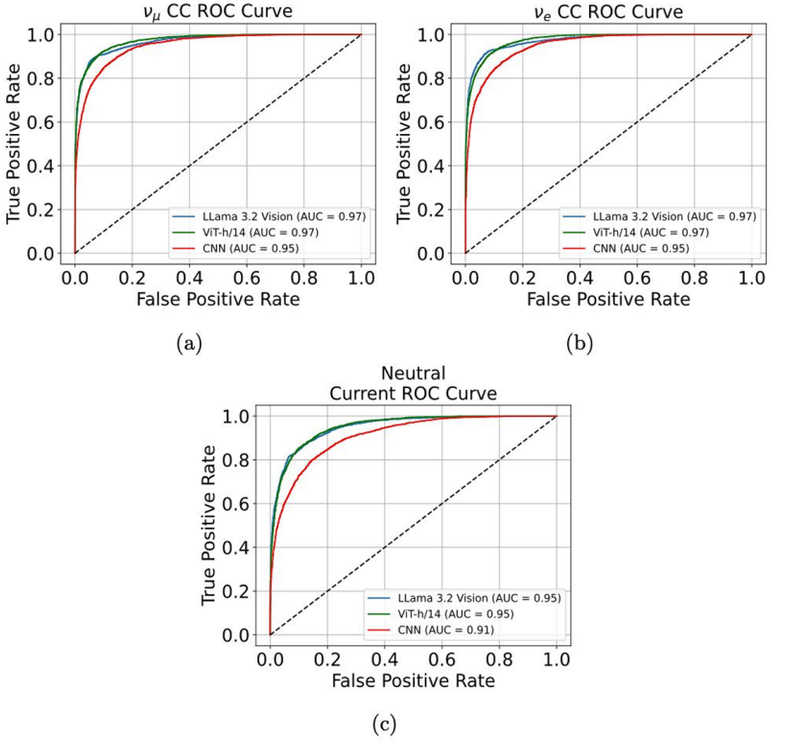
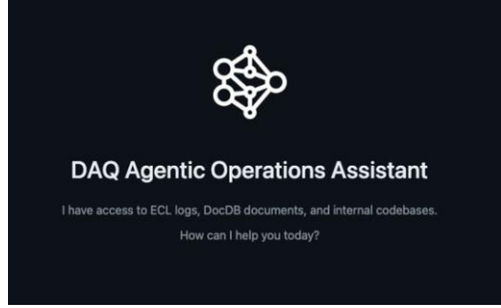


Figure 11: ROC curves for each class (a) ν_μ CC, (b) ν_e CC, and (c) NC comparing performance between the finetuned LLaMA 3.2 Vision and the CNN.

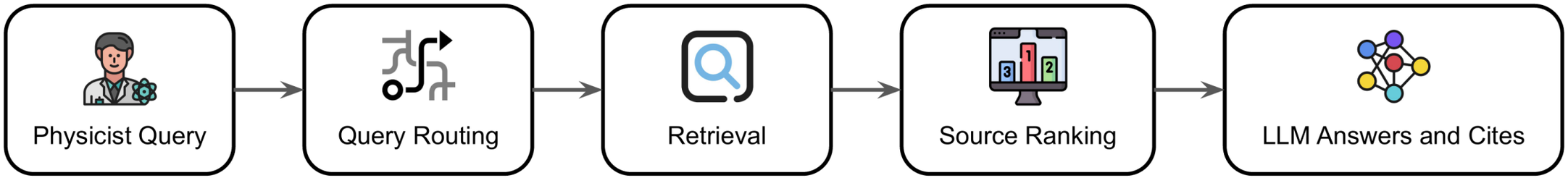
[Dikshant Sagar, Vision-Language Models for HEP Applications]
[2509.08461] Adapting Vision-Language Models for Neutrino Event Classification in High-Energy Physics

Our Work



Deploying Agentic RAG-LLM for DUNE DAQ Operations

LLM-powered assistant grounded in real experiment and data sources with little to no hallucinations from thin air. Using NOvA as a testbed.



NOvA ECL

NOvA's Electronic Collaboration Log - thousands of shift reports, anomaly notes, run-start/stop records, detector conditions. *Answers:* 'When did we last see beam spill issues?', 'What was the FD status in Oct 2023?'

Structured Logs | Run Metadata | Shift Reports

DocDB Documents

Internal physics notes, technical design documents, calibration reports, and analysis memos from NOvA and DUNE DocDB repositories. *Answers:* 'What's the current ν_e selection efficiency?', 'Find the latest FHC horn current calibration note.'

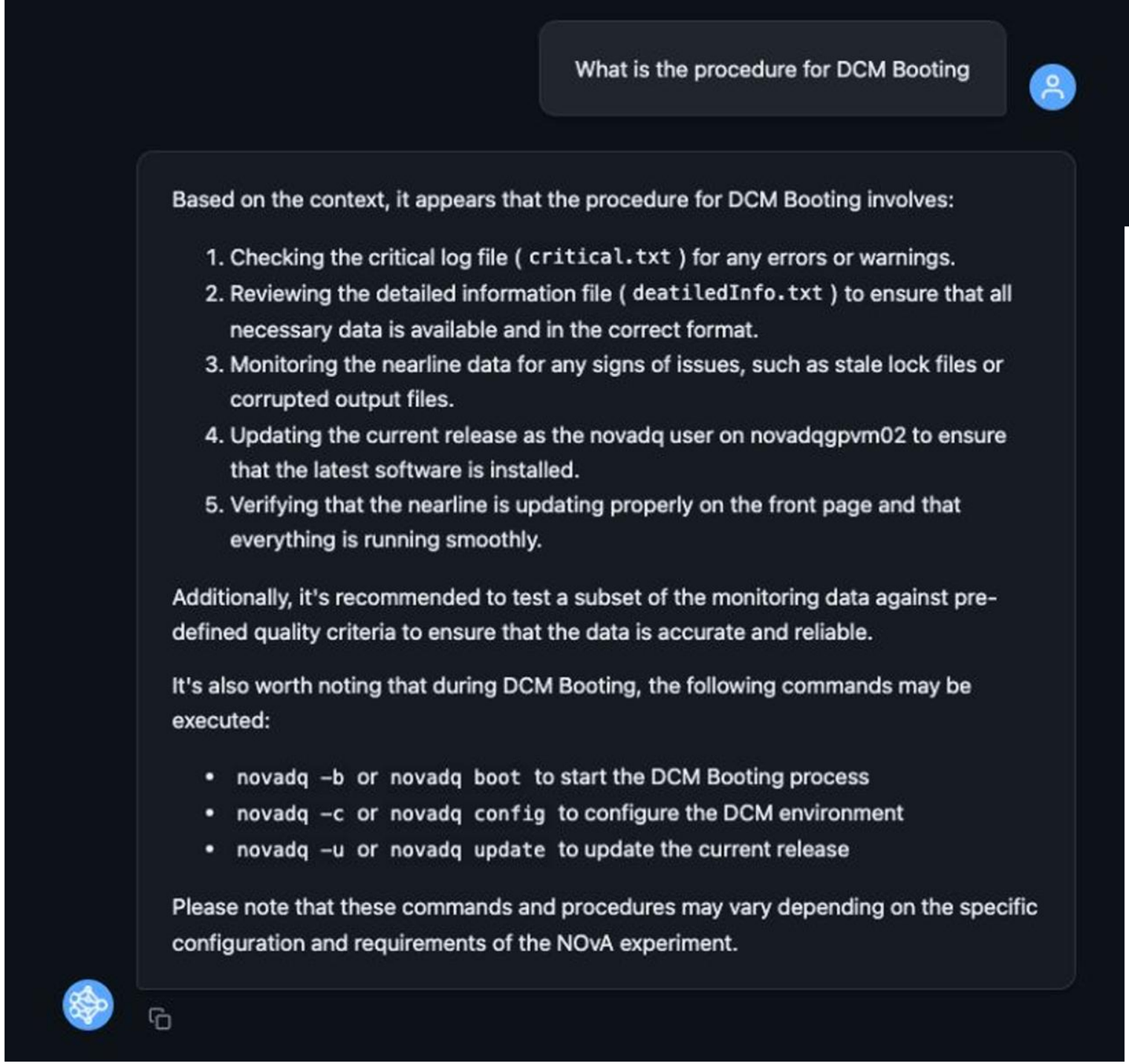
Technical Notes | Analysis memos | Design Docs

Internal Codebases

NOvAsoft / LArSoft source code, reconstruction modules, and analysis scripts. *Answers:* 'How does the Kalman tracker handle broken tracks?', 'Where is the beam timing correction applied?', 'Show me the CVN input variables.'

NOvASoft | LArSoft | Analysis Scripts

Key property: every answer is grounded in a cited source document - physicists can click through to the original ECL entry, DocDB note, or code line. No confabulation.


What is the procedure for DCM Booting
DAQ Agentic Operations Assistant
I have access to ECL logs, DocDB documents, and internal codebases.
How can I help you today?
Based on the context, it appears that the procedure for DCM Booting involves:
1. Checking the critical log file (critical.txt) for any errors or warnings.
2. Reviewing the detailed information file (deatiledInfo.txt) to ensure that all necessary data is available and in the correct format.
3. Monitoring the nearline data for any signs of issues, such as stale lock files or corrupted output files.
4. Updating the current release as the novadq user on novadqgpvm02 to ensure that the latest software is installed.
5. Verifying that the nearline is updating properly on the front page and that everything is running smoothly.
Additionally, it's recommended to test a subset of the monitoring data against pre-defined quality criteria to ensure that the data is accurate and reliable.
It's also worth noting that during DCM Booting, the following commands may be executed:
• novadq -b or novadq boot to start the DCM Booting process
• novadq -c or novadq config to configure the DCM environment
• novadq -u or novadq update to update the current release
Please note that these commands and procedures may vary depending on the specific configuration and requirements of the NOvA experiment.

INFRASTRUCTURE

Infrastructure Requirements

What DUNE needs to deploy LLMs/VLMs at Scale

Compute Resources

- Minimum: 8× A100 80GB or H100 nodes for fine-tuning
- Inference cluster: 4–16 GPUs depending on throughput target
- Leverage NERSC Perlmutter + Fermilab Wilson Cluster
- Estimated GPU-hours/year: 50K–200K (scope-dependent)
- Cloud (AmSC) as fallback for large training runs

Storage & Networking

- Model registry: ~5–20 TB for checkpoints and versions
- Dataset storage: 50–500 TB for training corpora
- High-bandwidth NVMe for fast data loading during training
- Low-latency network path for online detector inference
- CVMFS / dCache for distributing model weights to sites.

Software Stack

- Hugging Face Transformers + Accelerate for fine-tuning
- vLLM or TGI (Text Generation Inference) for serving
- Weights & Biases for experiment tracking
- Docker containers for HPC portability
- LArSoft / dunereco plugin interface for reconstruction chain

COST

Cost & Resource Estimates

Order-of-magnitude planning figures for the DUNE AI/ML Program

50-200K
GPU-hours / year
Offline batch inference

5-50K
GPU-hours
Periodic fine-tuning runs

~500 TB
Storage needed
Models + training datasets

4-16
A100/H100 nodes
Dedicated Inference Cluster

Dikshant Sagar, Vision-Language Models for HEP Applications

RISKS

Challenges & Risks

What could go wrong and how to mitigate it

Physics Fidelity — HIGH

LLMs/VLMs lack calibrated uncertainty. Overconfident predictions could bias oscillation measurements. Require uncertainty quantification.

MC-to-Data Domain Gap — HIGH

Models trained on Geant4 sim may fail on real FD data due to uncaptured detector effects. Domain adaptation or physics-informed augmentation required before deployment.

Interpretability — MED

Foundation models are opaque. DUNE publications require explainable decisions. Need explanations with attention visualization, saliency maps, etc.

Reproducibility — MED

Autoregressive sampling is non-deterministic by default. Models in the reconstruction chain must produce reproducible results. Requires seeded inference pipelines.

Model Staleness — LOW

Detector conditions evolve (noise, gains, dead channels). Models trained on early-run data will degrade. Need continuous monitoring + periodic retraining cadence.

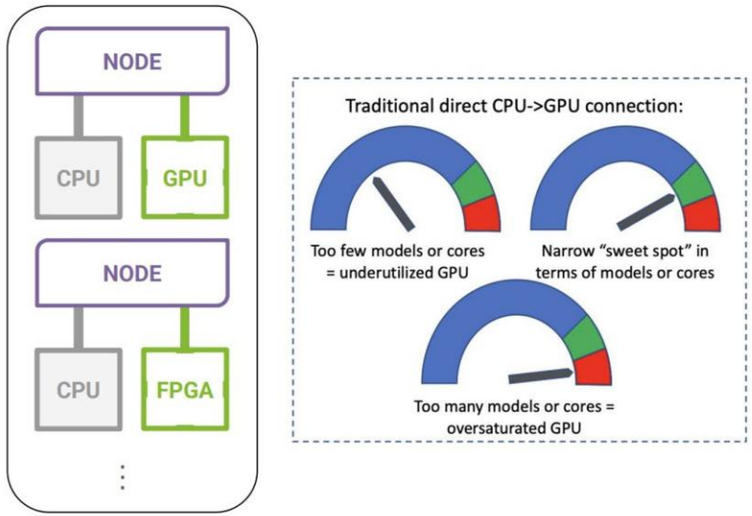
Regulatory/Export Control — LOW

Some LLM weights carry export control considerations. Ensure compliance with DOE and Fermilab security policies, especially for international DUNE collaborators.

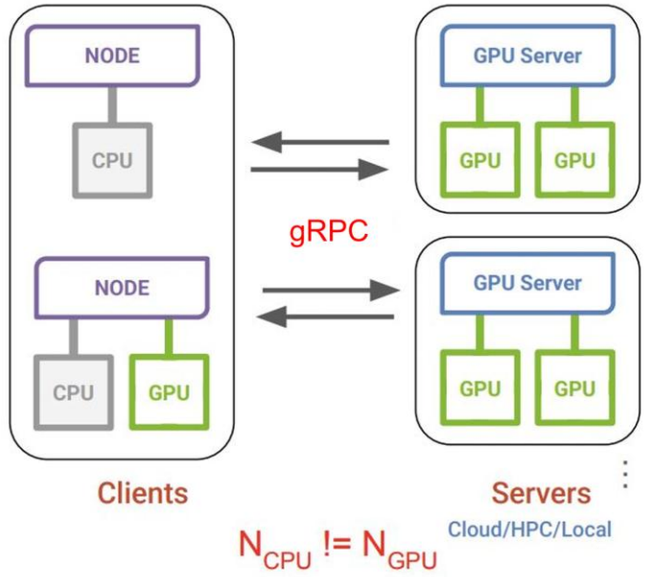
Inference as a Service

- Role of ML algorithms in HEP is growing, so does their computational share in workflows
- It's typical for trained models to run in production for years, accumulating high computational costs at inference stage

Use of coprocessors (GPU, FPGA, etc) is a must, but they are scarce and expensive, we need to use them efficiently



- Direct CPU → GPU connection inefficient
  - Wastes resource if inference load not known in advance

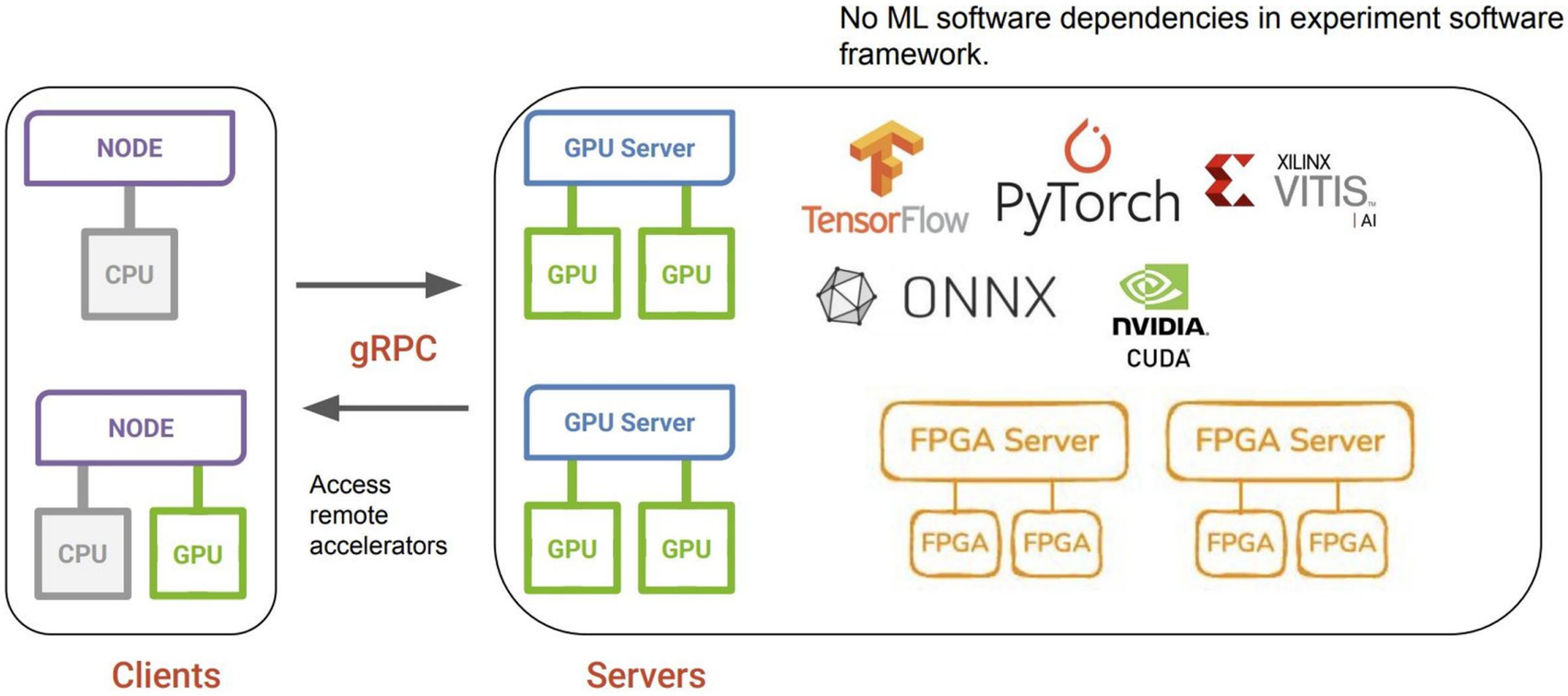


- IaaS provides a flexible, alternative deployment scheme where machines with coprocessors host an inference server and remote clients send inference requests via network connections
  - Allows to dynamically optimize resource usage

Meghna Bhattacharya, Accelerated Inference

Inference as a Service

- Decouples LArSoft-based reconstruction/analysis framework from specific ML software dependency

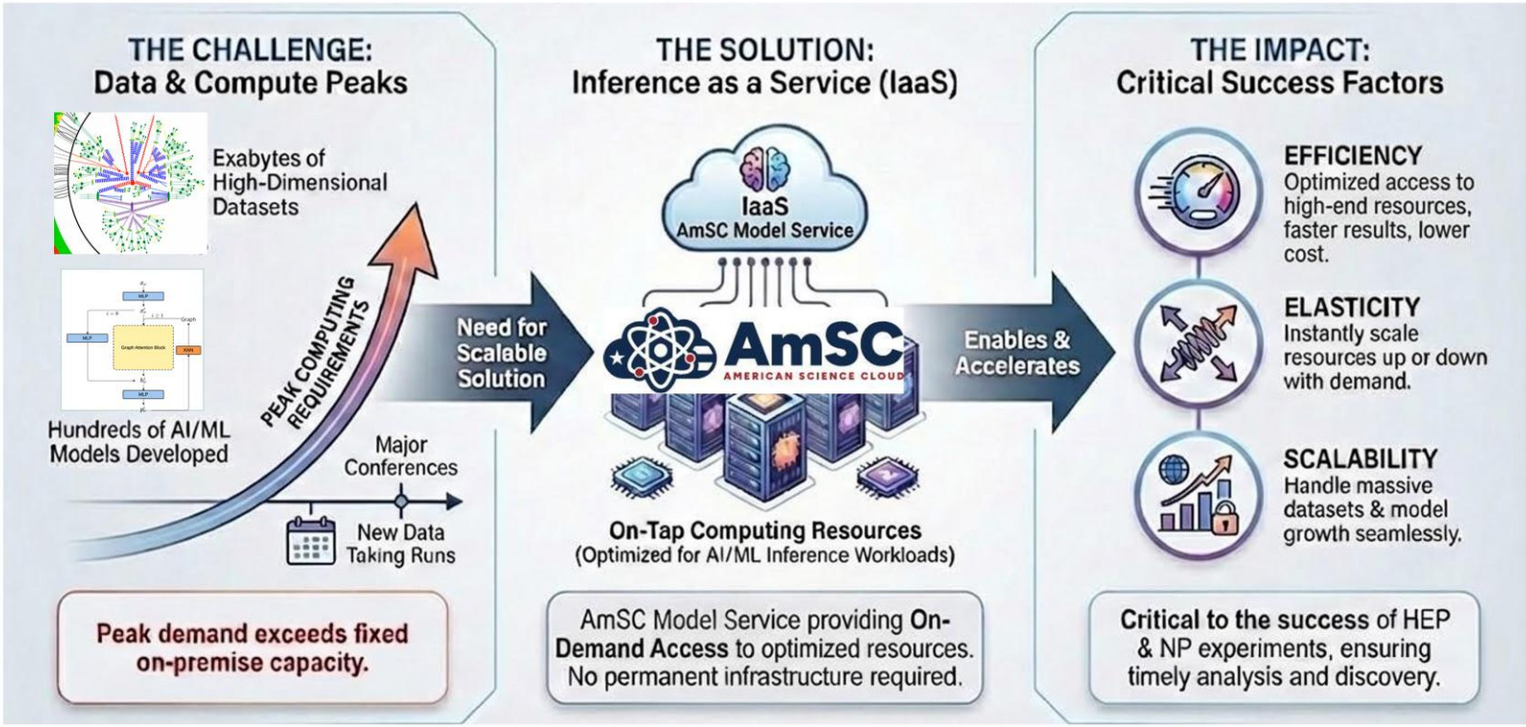


With IaaS: experiment software doesn't need to maintain builds, dependencies, version updates for PyTorch or TensorFlow

Meghna Bhattacharya, Accelerated Inference

Genesis Mission

- American Science Cloud
  - Scientific User Facility (SUF)



- Two LArTPC workflows along with ATLAS production workflows
  - Nugraph using MicroBooNE Opendata (https://amsc.fnal.gov:2880/amsc/neutrino/uboone/)
  - Supernova reconstruction pipeline with a ML-based raw data reduction (https://amsc.fnal.gov:2880/amsc/neutrino/dune/) restricted access ~ 30 TB low energy data
- Data lives in Fermi Data Platform (FDP), Genesis Mission's Infrastructure Partner
  - Many thanks to Steve Timm and the FDP team!
- Near-term Deliverable: IaaS demonstration in mid-May

Meghna Bhattacharya, Accelerated Inference

Partially digested summary

Callum Wilkinson
LBNL

- Disclaimer:
  - There are still a lot of unknowns so don't expect too much from this talk!

Callum Wilkinson, a_sort_of_summary.pdf

Partially digested summary

- A huge breadth of work covered, but it struck me that there were three strands to the talks:
  - **“Core” work:** AI/ML approaches to sim/reco/analysis workflows
  - **“Operations” work:** LLMs, DUNE-pro agent, …
  - **“Vision” work:** FMs, diff-sim, Genesis, …
- It seems to me that each has different computing needs, and probably need different “policy” approaches
- In particular, “vision” work might need a light-touch, but there needs to be a defined process for managing the “vision” to “core” transition

Callum Wilkinson, a_sort_of_summary.pdf

Resource needs/usage

- Two issues came up several times:
  - Disk, particularly training samples, but surprises from PWG plans
  - GPU access, both for training AI/ML and other needs
- Generally, concerns raised that some current work uses private resources, and results/files aren't appropriately documented
- If these are resources which DUNE has access to, but DUNE work isn't attributed to DUNE, it could compromise future access
- Aaron raised the importance of FAIR principles (Findability, Accessibility, Interoperability and Reuse of digital assets)

Callum Wilkinson, a_sort_of_summary.pdf

FAIR AI/ML

- Complicated by the progression of much AI/ML research: experiment independent tests → experiment-specific adaptation → production quality results *(“vision” → “core”)*
- Policy needed to make expectations clear, but worth discussing further. Do we all agree on what needs to be stored?
- E.g., do training samples need to be stored? Is it sufficient to make it possible to reproduce training samples?
- On one hand, training samples can and should be re-used for some cases → SPINE/other reco flat training samples
- On the other, a pre-trained model which uses non-DUNE data might boost performance… what do we do with that case?

Callum Wilkinson, a_sort_of_summary.pdf

FAIR AI/ML

- SPINE approach seems like a good starting point
- Would more info be needed to retrain the model if we needed to? E.g., if the detector conditions change?
- Policy needed?

Portability

SLAC NATIONAL ACCELERATOR LABORATORY

SPINE **dependencies packaged** in an Apptainer/Docker image here

SPINE is tagged and released on PyPI (pypi.org/project/spine)

Each training process yields **one inference config + one weight file**

- **Inference configuration** shared through spine-prod (tagged)
  - Ships with tagged version of SPINE for reproducibility
  - Ships with trivial submit.py for end users (slurm-only for now)
- **Weight files (~110 MB per file)** hosted publicly
  - Tagged, download + chesum for validation by spine-prod

Documentation on ReadTheDocs: spine.readthedocs.io

SPINE @ DUNE, F. Drielsma 35

Callum Wilkinson, a_sort_of_summary.pdf

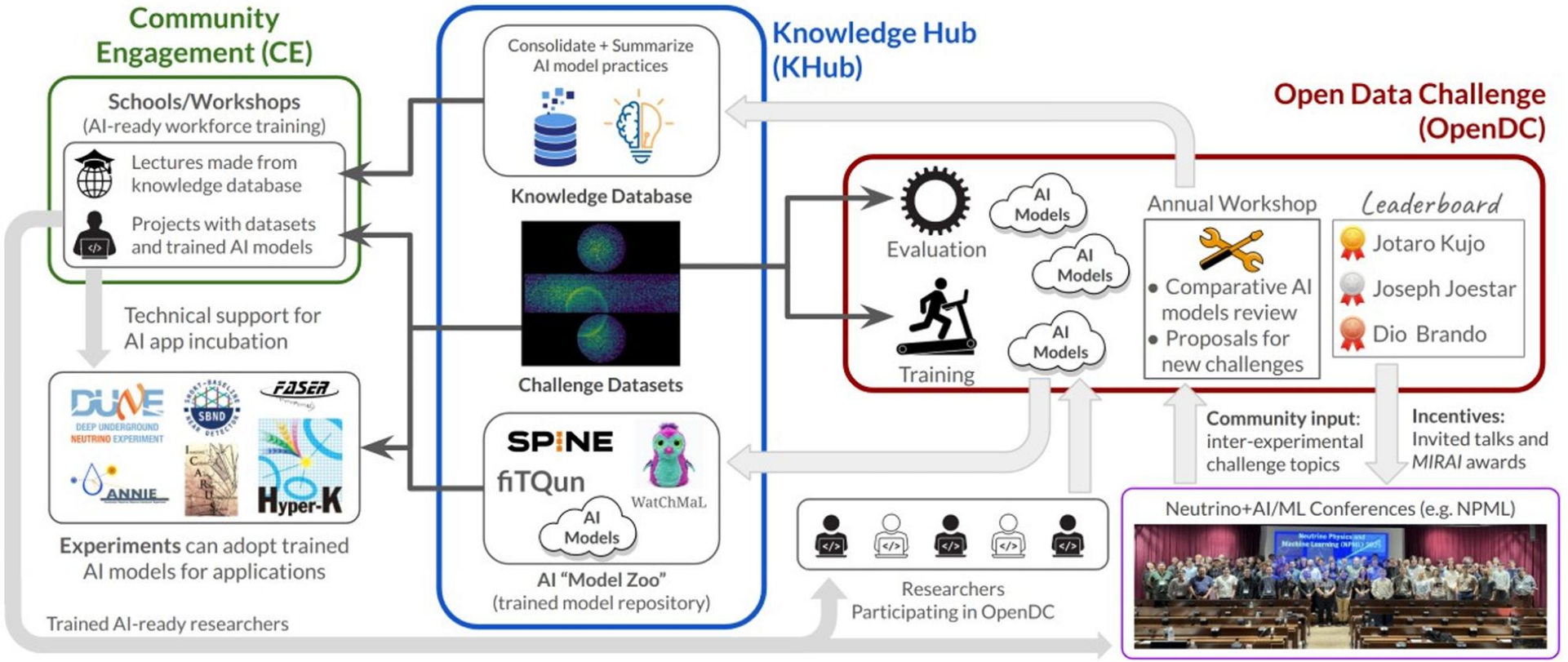
Benchmarking

- IMO one additional requirement should be the ability to reproduce performance plots → e.g., benchmark models
- Ideally, datasets for benchmarking should be used by different approaches to the same problem
- (E.g., you can’t retrain many image classification models, but you can check how they perform on ImageNet etc)

Callum Wilkinson, a_sort_of_summary.pdf

Open datasets

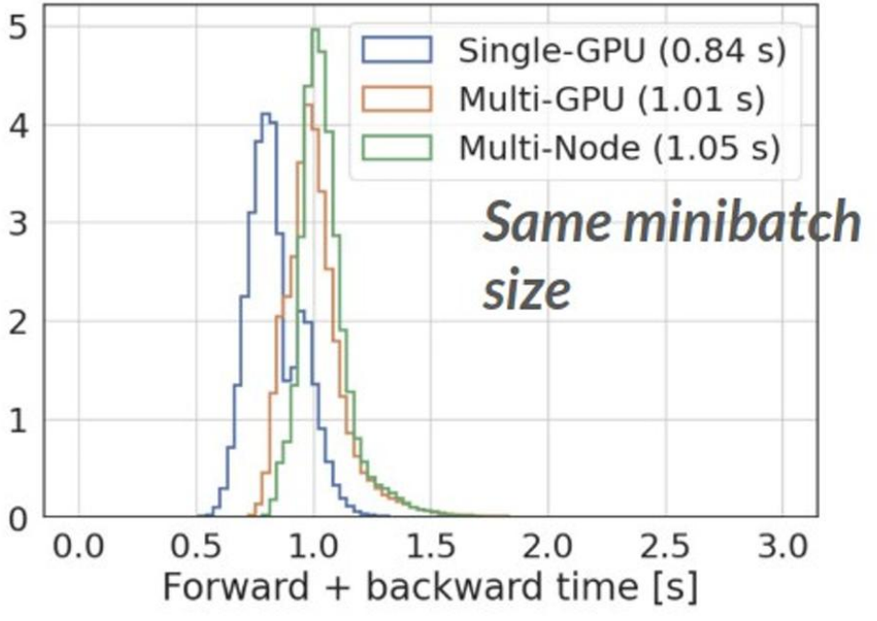
- Open datasets and common benchmarks could bring in more AI/ML expertise and help provide training tools and resources
- Community efforts in this direction have started, there's an opportunity to lead the way as a large collaboration

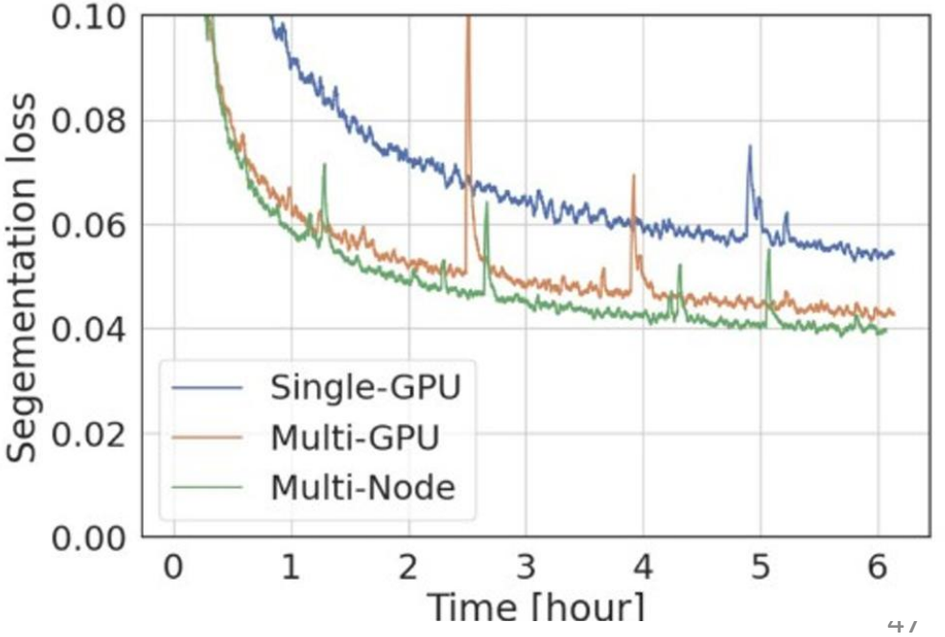


Callum Wilkinson, a_sort_of_summary.pdf

Reproducible training?

- For training larger models, there can be performance benefits from exploiting site-specific features
- E.g., many parallel nodes with fast interconnect, larger than normal VRAM, tensor cores, others…
- Should training be treated separately from inference for policy/planning purposes? Should it use standard production tools?





Callum Wilkinson, a_sort_of_summary.pdf

GPU portability

- Many outstanding questions regarding GPU needs:
  - Training vs production
  - Interface through Phlex → requirements may need revisiting!
  - CPU inference or IaaS (triggers, Pandora, others …)
  - NVIDIA specific features – Opticks
  - Opportunities for external help: CAAR at ORNL, Genesis, HEP-CCE PAW, …

    ...but potential problems with NDAs etc…
- Was this captured by previous surveys? Need for more detailed consultation/discussion?

Callum Wilkinson, a_sort_of_summary.pdf

“Vision” work

- This is probably the most problematic from a computing planning point of view:
    - Straddles line between collaboration and individual research, (similar to phenomenological/theory work
    - Clearly important for maintaining vitality of AI/ML group
    - Potential for new funding/responding rapidly to new opportunities
    - But hard to plan resource allocation for → *should we?*
    - What about work which crosses experimental boundaries?
    - Need a strategy for bringing “on-shell”
- Thoughts? Violent objections to this characterization?

Callum Wilkinson, a_sort_of_summary.pdf

Ongoing co-ordination between groups?

- Need for dedicated AI/ML representation in the CRAB
- “AI/ML tools” in the computing organization? Several important questions:
  - Which packages need support? And what that entails
  - GPU portability layer
  - Exploring other GPU resource options
  - Phlex implications
  - Production
  - ...
- Co-co-ordination of the “operations” work? Potential for targeted funding requests for this piece, outside general AI/ML funding calls
- Regular cross-group meetings like this?
- Co-ordinated training efforts?

Callum Wilkinson, a_sort_of_summary.pdf

Thanks to Aaron, Ilker and Rice for hosting us!

And to everyone for the work and erudite discussions!

Callum Wilkinson, a_sort_of_summary.pdf