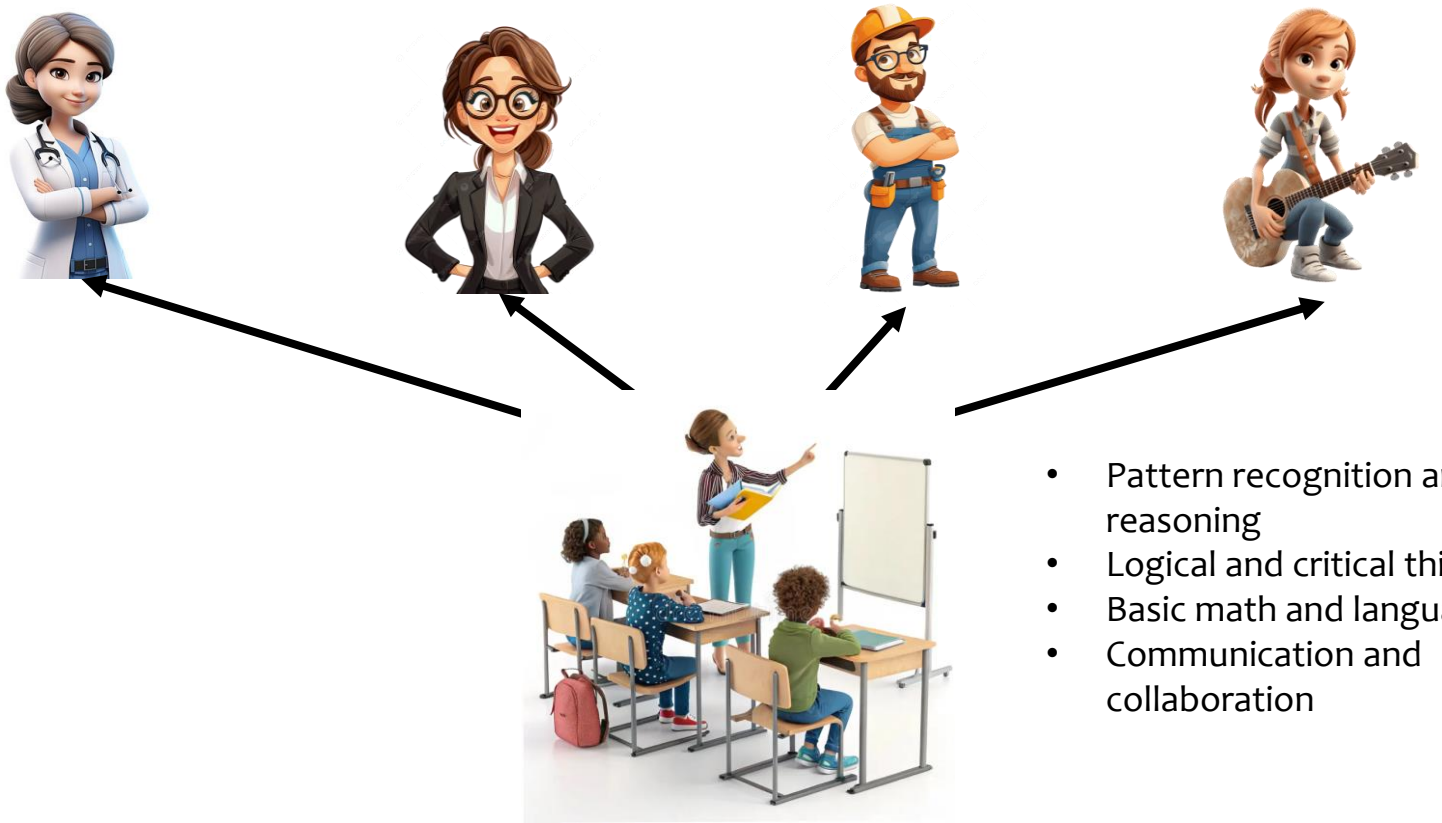
Beyond Task-Specific Models

Foundation Models as a New Paradigm for Experimental Physics

TREASURE: Tokenizing HEP Collider Data for AI Discovery

Yi Huang
04-27-2026

Foundation vs Task-Specific Human



- Pattern recognition and rule reasoning
- Logical and critical thinking
- Basic math and language skills
- Communication and collaboration

What is Wrong of Task-Specific ML

Expensive Labels

Every task demands its own labeled dataset. The labels can be expensive and slow to produce. **Example:** Particle accelerators produce hundred and thousand particles at each collision. Annotate them would be a tremendous amount of work.

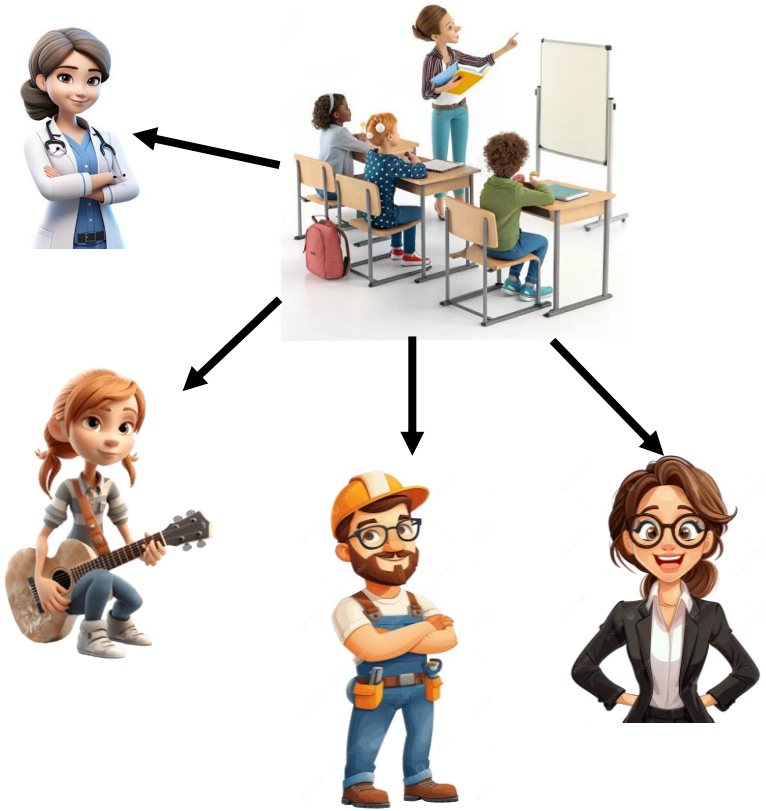
Unreusable Algorithm

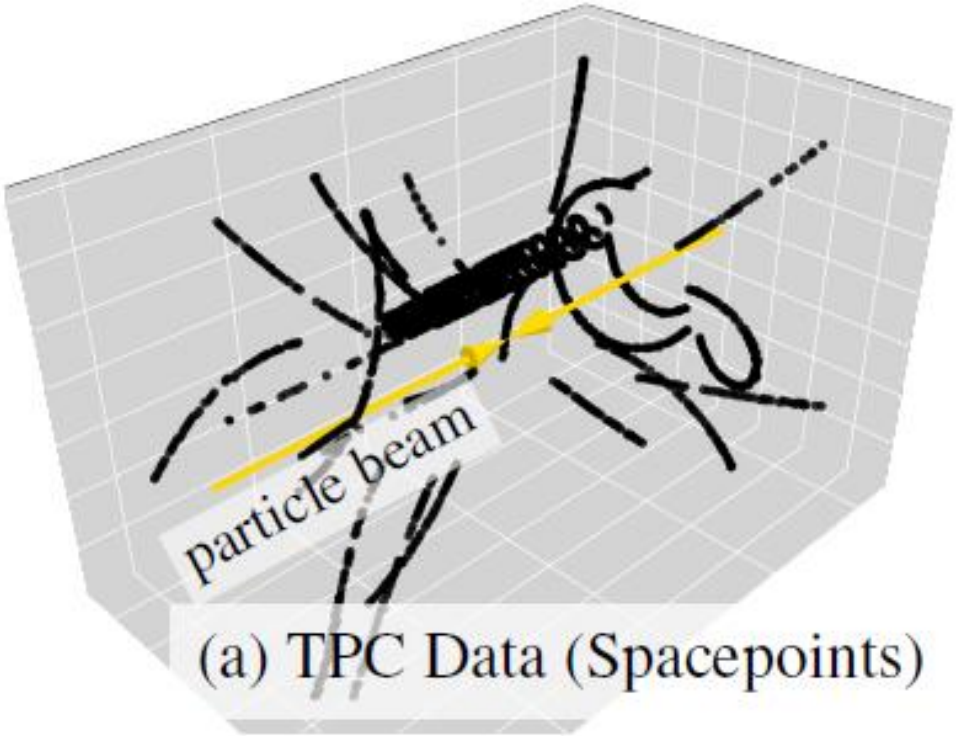
We will have to design a deep learning algorithm for each task. Each model must be trained, tuned, maintained individually. Overfitting may also happen easily for task-specific algorithms if the complexity of the algorithm and that of the problem do not match.

This is NOT how understanding works

If humans first learn more general concepts before specialize in tasks, why should artificial intelligent do the opposite. An AI model that understands physics can be trained to specialize in any task with physics as the underlying rule, but the opposite is not true.

Foundation vs Task-Specific ML





CONCEPT

What Is a Foundation Model?

Analogy

Training in logic & critical thinking sharpens reasoning across mathematics, economics, and political science alike — the general capability lifts all specific applications.

An ML model is a foundation model if it is trained to develop a holistic understanding of data that is transferable across many downstream tasks.

Key mechanism:

Self-supervised learning (SSL) — withhold or distort part of the data, challenge the model to reason about what is missing from what remains.

METHODS

A Landscape of Self-Supervised Strategies

Masked Reconstruction

Hide tokens/patches; predict missing content.
BERT, MAE.

Contrastive Learning

Pull augmented views of same sample together,
push different samples apart.
SimCLR, MoCo.

Prototype Distillation

Teacher assigns target representation on full data;
student predicts them on masked/augmented views.
DINO, Panda.

Forecasting

Predict future or context states
from partial observations.
GPT-style, FM4NPP

All share the same philosophy: structure in the data is rich enough to be learned from context alone.

Masked Reconstruction — Learn by Filling in the Blanks

Core idea

Randomly mask (hide) a portion of the input. Train the model to reconstruct the missing content from the visible context alone.

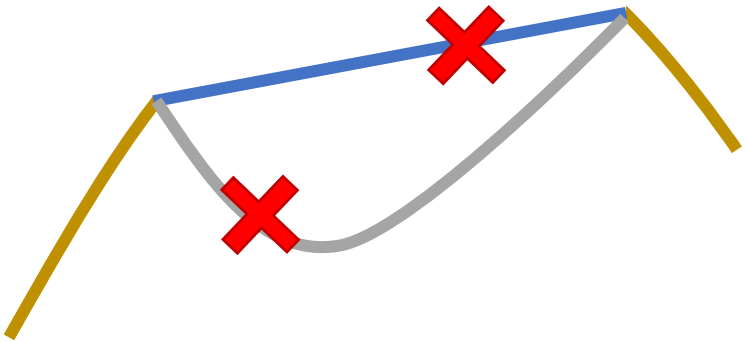
Why it works

To predict the missing parts, the model must build a deep understanding of structure, context, and relationships in the data.

Examples ML Models

- BERT — mask tokens in text
- MAE — mask patches in images

The _____ cat secretly ordered a pizza at 3 a.m. and blamed the _____.



Contrastive Learning — Similar Together, Different Apart

Core idea

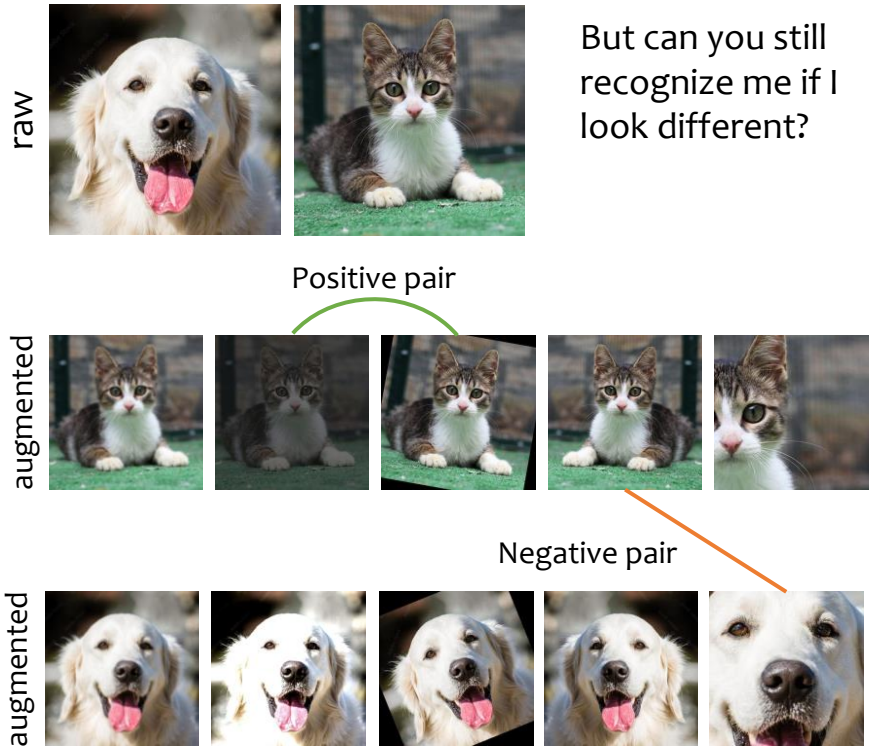
Create augmented views of the same sample. Train the model so these views land close together in embedding space, while views of different samples are pushed apart.

Why it works

The model learns invariances — representations that are stable under noise, rotation, masking — which are precisely the features that matter for physics.

Examples ML Models

- SimCLR — image augmentation pairs
- MoCo — momentum encoder + queue



Prototype Distillation — Learn by Agreeing with Yourself

Core idea

A teacher model that learns and generates representation from full data. A student model learn to produce similar data representation from a masked or distorted view.

Why it works

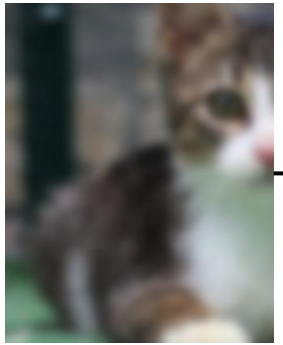
If a model can still produce correct and detailed understand from partial and/or distorted data, it must understand the hidden correlation in data well.

Examples ML Models

- DINO / DINOv2 — vision transformers
- SwAV — clustering with prototypes
- **Panda — particle trajectory prototypes in LArTPC**

Teacher model

A young tabby cat with white chest sitting on the ground in the Sphinx pose. Its eyes are widely open with narrow pupils…



Student model

A cat with white chest sitting on the ground. Its eyes are widely open.

Teacher model at time step i $= a$ Student model at time step i $+ (1 - a)$ Student model at time step $i - 1$

Forecasting — Learn by Predicting What Comes Next

Core idea

Process a sequence of tokens left-to-right (or in some causal order). At each step, predict the next token from all previous context — no labels needed, just the data itself.

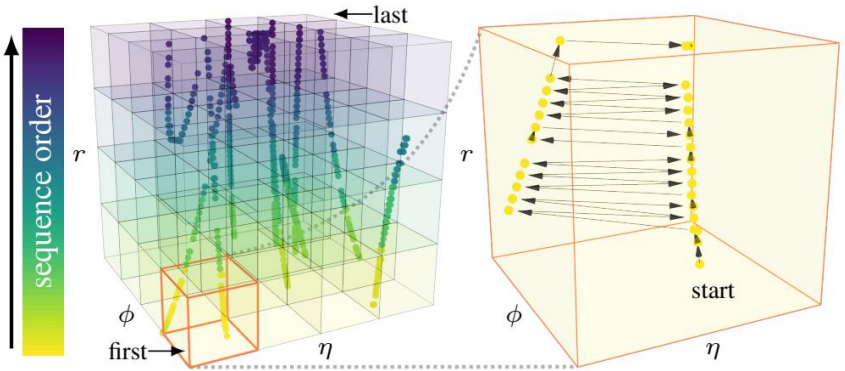
Why it works

Every position in the sequence becomes both an input and a prediction target. Long-range dependencies, local patterns, and global structure all emerge from the same objective.

Examples ML Models

- GPT / LLMs — next token prediction
- **FM4NPP — predict location of the next detector hit**

- I finally taught my goldfish how to code, but now it only writes programs in _____.
- The penguin built an AI model to catch fish, but it kept crashing whenever it saw a _____.



SUMMARY

Four Strategies, One Philosophy

Masked Reconstruction	Contrastive Learning	Prototype Distillation	Predictive Coding
Signal Predict hidden tokens	**Signal** Align augmented views	**Signal** Match teacher cluster labels	**Signal** Predict next in sequence
Captures Local structure & correlations	**Captures** Invariance to transformations	**Captures** Emergent semantic categories	**Captures** Long-range dependencies
e.g. BERT · MAE	**e.g.** SimCLR · MoCo	**e.g.** DINO · Panda	**e.g.** GPT · FM4NPP

All four: withhold part of the data · challenge the model to reason from what remains · no labels required

CASE STUDY I

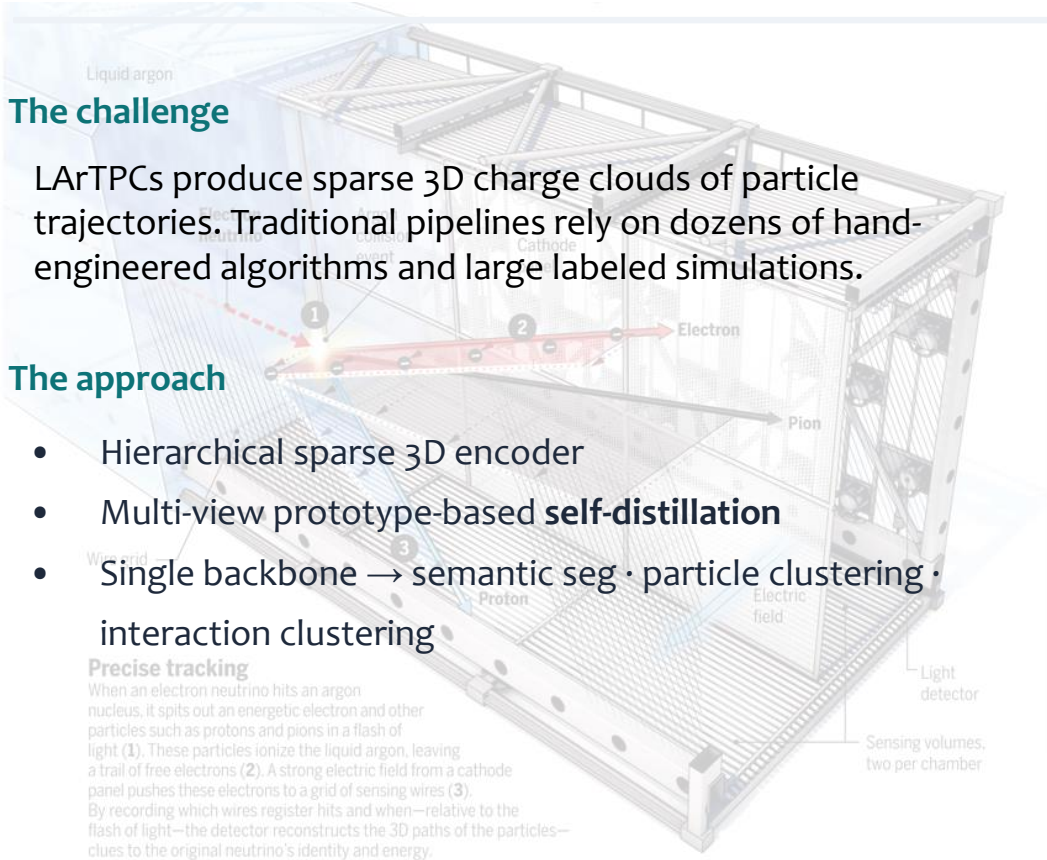
Panda — Self-Distillation for LArTPC Data

The challenge

LArTPCs produce sparse 3D charge clouds of particle trajectories. Traditional pipelines rely on dozens of hand-engineered algorithms and large labeled simulations.

The approach

- Hierarchical sparse 3D encoder
- Multi-view prototype-based **self-distillation**
- Single backbone → semantic seg · particle clustering · interaction clustering



Key Result

1,000×

fewer labels to match state-of-the-art segmentation

Young & Terao, arXiv:2512.01324

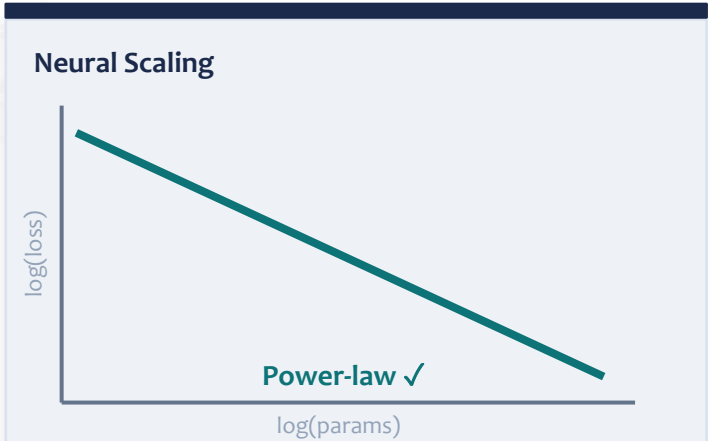
FM4NPP — Scaling Laws for Nuclear & Particle Physics (more details to come)

The challenge

sPHENIX detector at RHIC produces sparse, high-dimensional collision events. Do scaling laws — proven in NLP — carry over to physics data?

The approach

- 11M+ particle collision events;
- Models up to 188M parameters; scaling evaluated across size, data & compute
- **Forecasting** self-supervised training method
- Frozen backbone + task adapter → track finding, particle ID, noise tagging

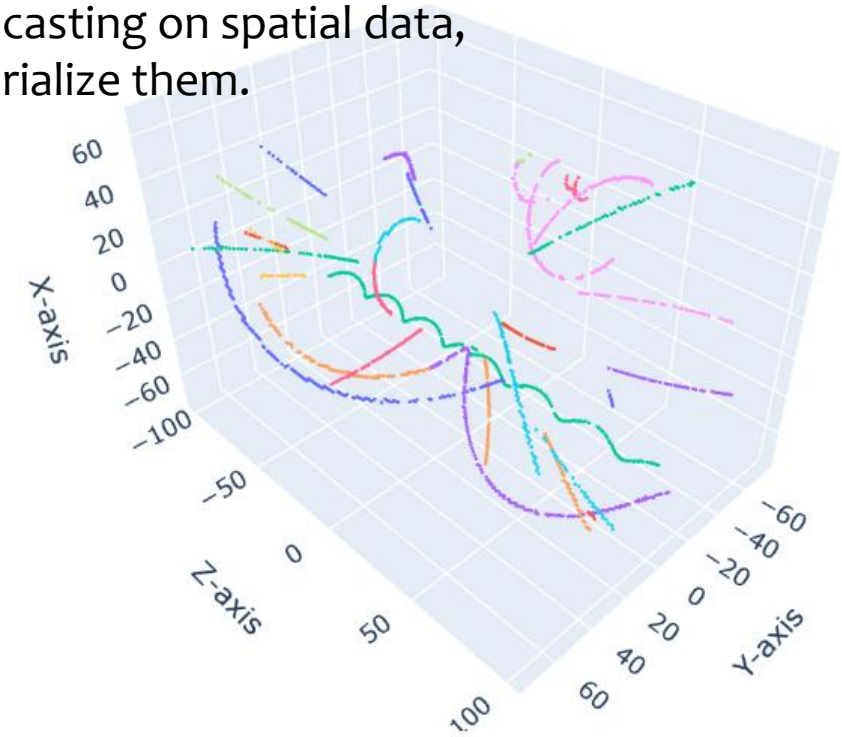


FM consistently outperforms task-specific baselines across all downstream tasks — with frozen weights.

Park et al., arXiv:2508.14087

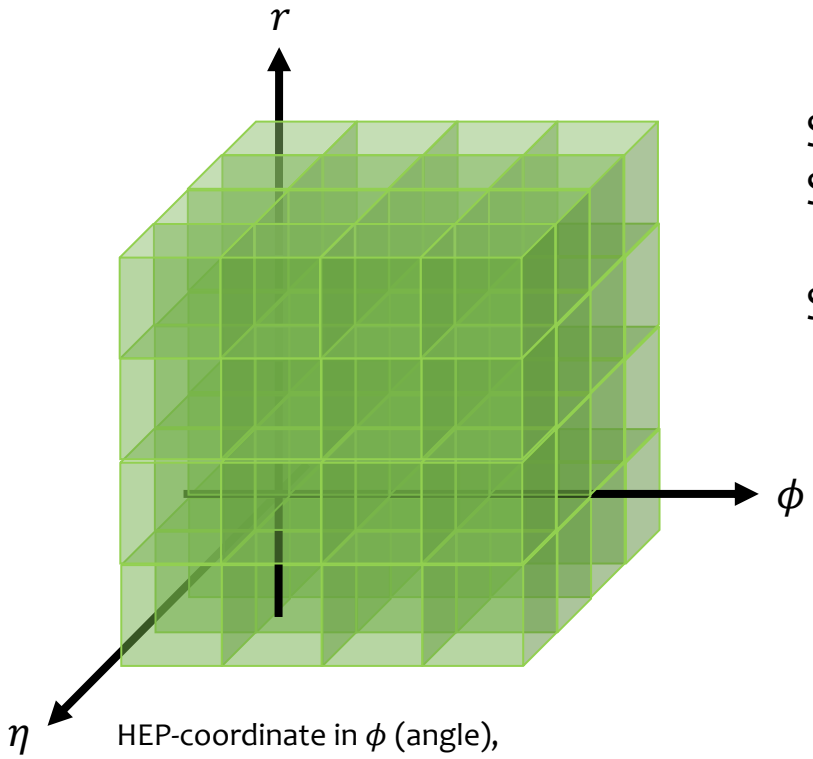
How Do We Do Forecasting for Spatial Data?

In order to do forecasting on spatial data,
we first need to serialize them.



An example input point cloud from p + p collision. The input is a list of points in (E, x, y, z).

How Do We Do Forecasting for Spatial Data?



HEP-coordinate in ϕ (angle), η (pseudo-rapidity), and radius

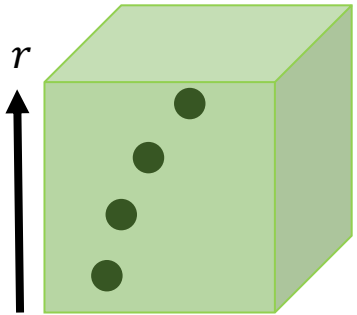
Step 1: Subdivide the space into blocks

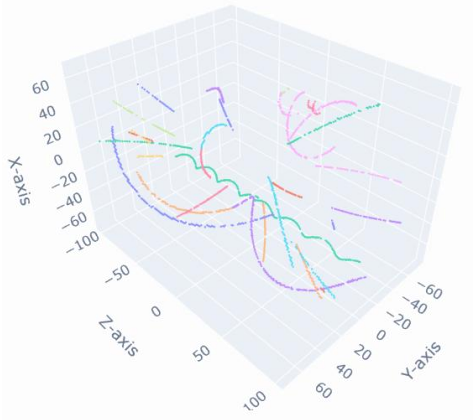
Step 2: Order the blocks in a given order

(for example, in ϕ, η, r order)

Step 3: Order the points inside a block

(for example, in radius)





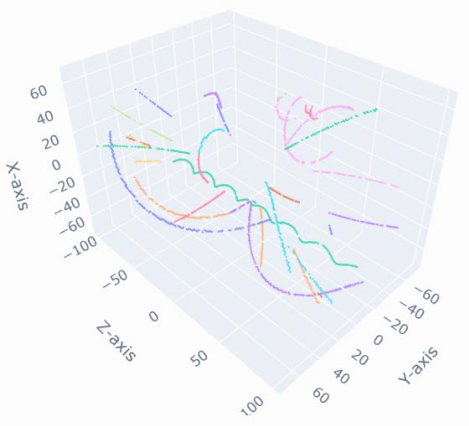
Forecasting Spatial Data

After this serialization process, all space points get a unique index in input, just like a word in a sentence.

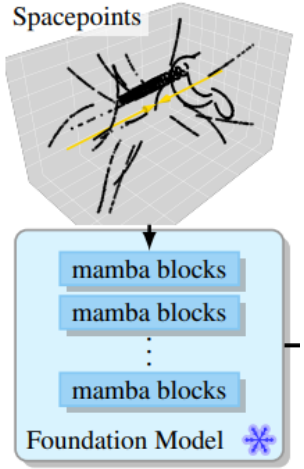
With this ordering, we can then apply any forecasting algorithms, such as Transformer-based algorithm, Mamba (a state-space model), to do forecasting.

What have encouraged the model to learn useful tokenization of the points?

A point's position could only be predicted accurately if points on the same track can be recognized by the model. This force the model to assign similar tokenization (representation/embedding) to points from the same track.



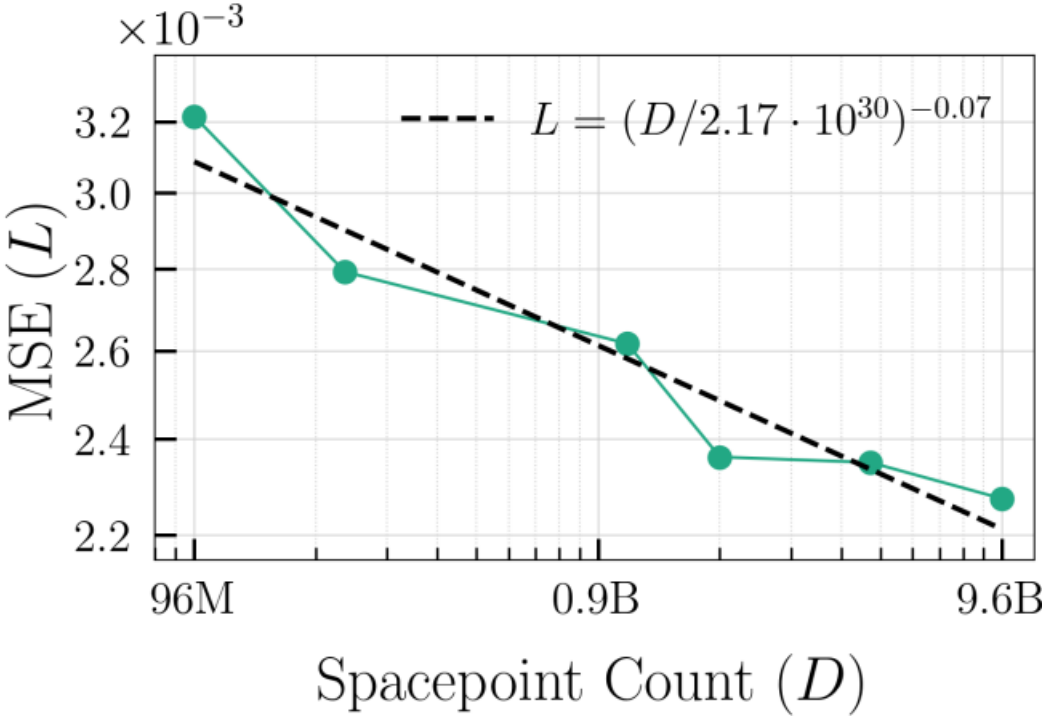
Fine-tuning for Downstream Tasks



forecasting pretraining

Fine-tuning: Train adapter models for downstream tasks

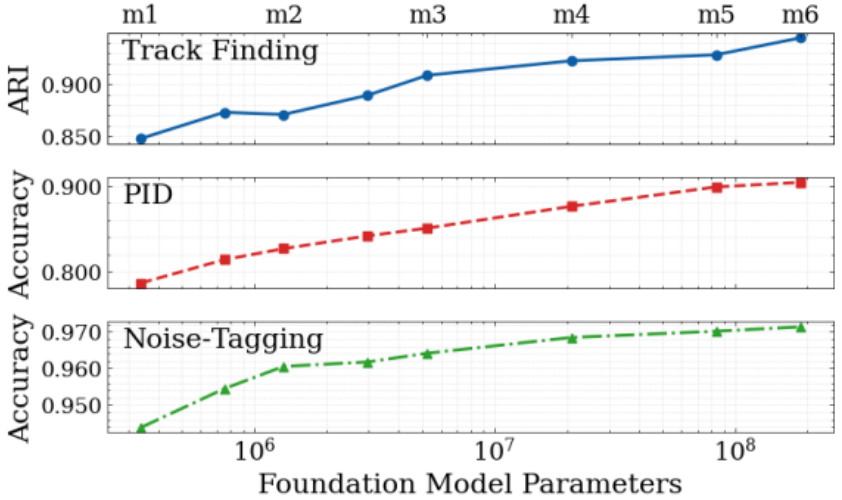
The More the Better?



Forecasting performance a function of input number of space points

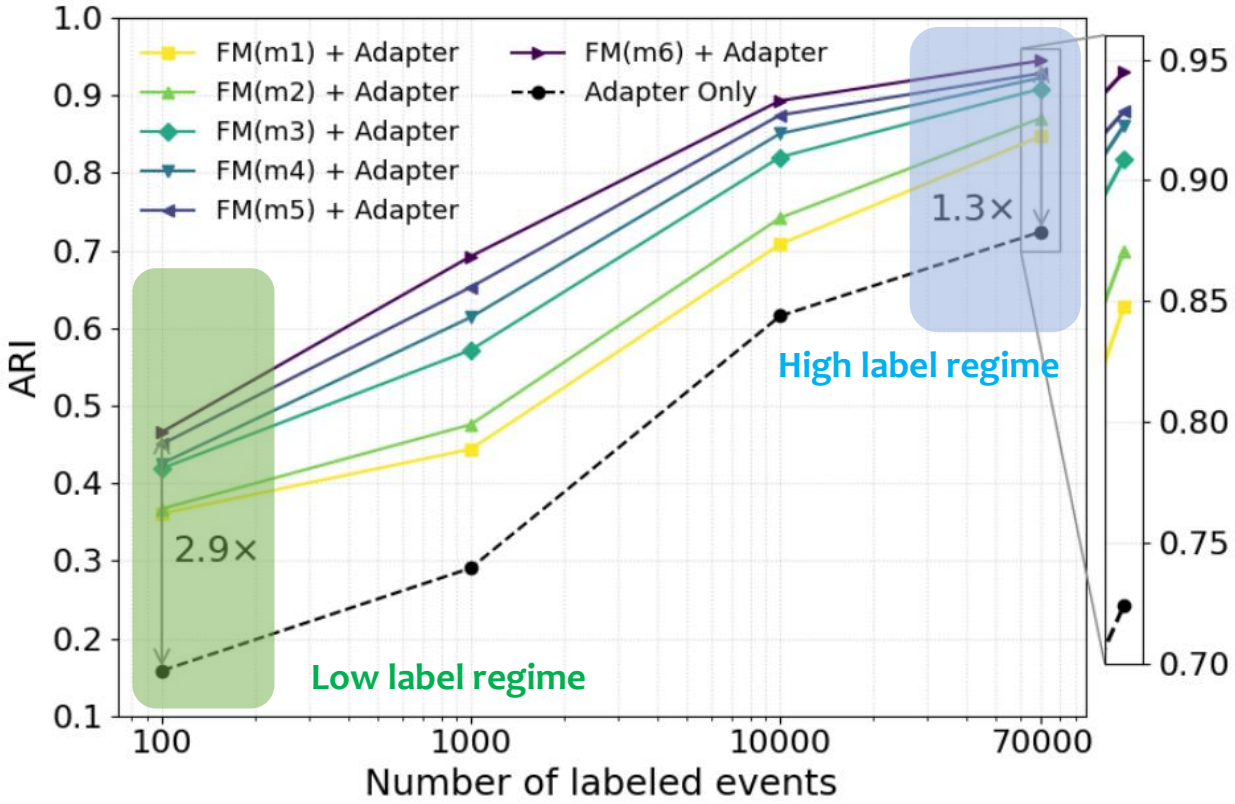
The Larger the Better?

	Model Sizes					
	m1	m2	m3	m4	m5	m6
Model Width	64	128	256	512	1024	1536
Model Params	0.34M	1.3M	5.3M	21M	84M	188M
	Compute Resources					
NVIDIA GPU	H100 80GB		A100 80GB			
Num GPUs	1	1	4	8	24	64
Train Hrs	10	12	20	32	50	72



Downstream task performance a function of model size

Label efficiency
(can we survive on few labeled data)



Track finding performance as function of labeled events

A Vision for Physics with Foundation Models

Shared and reusable backbone across multiple detectors and experiments

Labels become optional — self-supervised pretraining carries the weight.

Scaling laws suggest: more data + larger models = better physics, automatically.

The world is structured enough to be learned from context alone.

Thank You

FM4NPP
BNL and Columbia

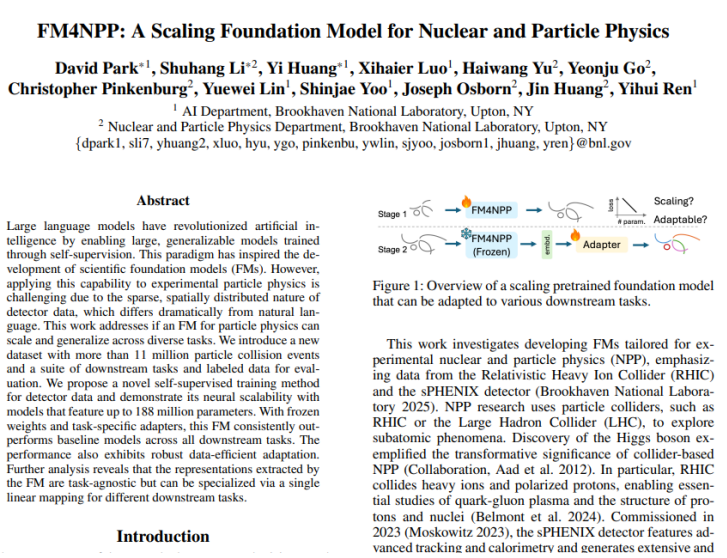
Park, Li et al.

arXiv:2508.14087 · Aug 2025

Scaling FM for nuclear & particle physics

Questions & Discussion

yhuang2@bnl.gov



FM4NPP: A Scaling Foundation Model for Nuclear and Particle Physics

David Park*[1], Shuhang Li*[2], Yi Huang*[1], Xihaier Luo[1], Haiwang Yu[2], Yeonju Go[2], Christopher Pinkenburg[2], Yuewei Lin[1], Shinjae Yoo[1], Joseph Osborn[2], Jin Huang[2], Yihui Ren[1]
[1] AI Department, Brookhaven National Laboratory, Upton, NY
[2] Nuclear and Particle Physics Department, Brookhaven National Laboratory, Upton, NY
{dpark1, sli7, yhuang2, xluo, hyu, ygo, pinkenbu, ywlin, sjyoo, josborn1, jhuang, yren}@bnl.gov

Abstract

Large language models have revolutionized artificial intelligence by enabling large, generalizable models trained through self-supervision. This paradigm has inspired the development of scientific foundation models (FMs). However, applying this capability to experimental particle physics is challenging due to the sparse, spatially distributed nature of detector data, which differs dramatically from natural language. This work addresses if an FM for particle physics can scale and generalize across diverse tasks. We introduce a new dataset with more than 11 million particle collision events and a suite of downstream tasks and labeled data for evaluation. We propose a novel self-supervised training method for detector data and demonstrate its neural scalability with models that feature up to 188 million parameters. With frozen weights and task-specific adapters, this FM consistently outperforms baseline models across all downstream tasks. The performance also exhibits robust data-efficient adaptation. Further analysis reveals that the representations extracted by the FM are task-agnostic but can be specialized via a single linear mapping for different downstream tasks.

Introduction

Figure 1: Overview of a scaling pretrained foundation model that can be adapted to various downstream tasks.

This work investigates developing FMs tailored for experimental nuclear and particle physics (NPP), emphasizing data from the Relativistic Heavy Ion Collider (RHIC) and the sPHENIX detector (Brookhaven National Laboratory 2025). NPP research uses particle colliders, such as RHIC or the Large Hadron Collider (LHC), to explore subatomic phenomena. Discovery of the Higgs boson exemplified the transformative significance of collider-based NPP (Collaboration, Aad et al. 2012). In particular, RHIC collides heavy ions and polarized protons, enabling essential studies of quark-gluon plasma and the structure of protons and nuclei (Belmont et al. 2024). Commissioned in 2023 (Moskowitz 2023), the sPHENIX detector features advanced tracking and calorimetry and generates extensive and