

Leveraging Chatbots in daily HEP work - Open Source and Commercial

G. Watts (UW/Seattle)

IML Working Group 2025-07-01



My work in this area is driven by:

- Overwhelmed by information – arxiv, conferences, workshops
 - There is stuff I want to know, but no longer have the time to look.
- ATLAS software is complex – can we do better?
 - Fascinated by this claim that AI can replace programmers

Note: This is all exploratory work! Nothing is in production, and I'm currently working alone

The last time I touched this code was ~yesterday.

It is all (very probably) out of date by now

I don't try to keep up!

I have to teach, mentor, physics!

“Traditional” RAG

Introduction to RAG

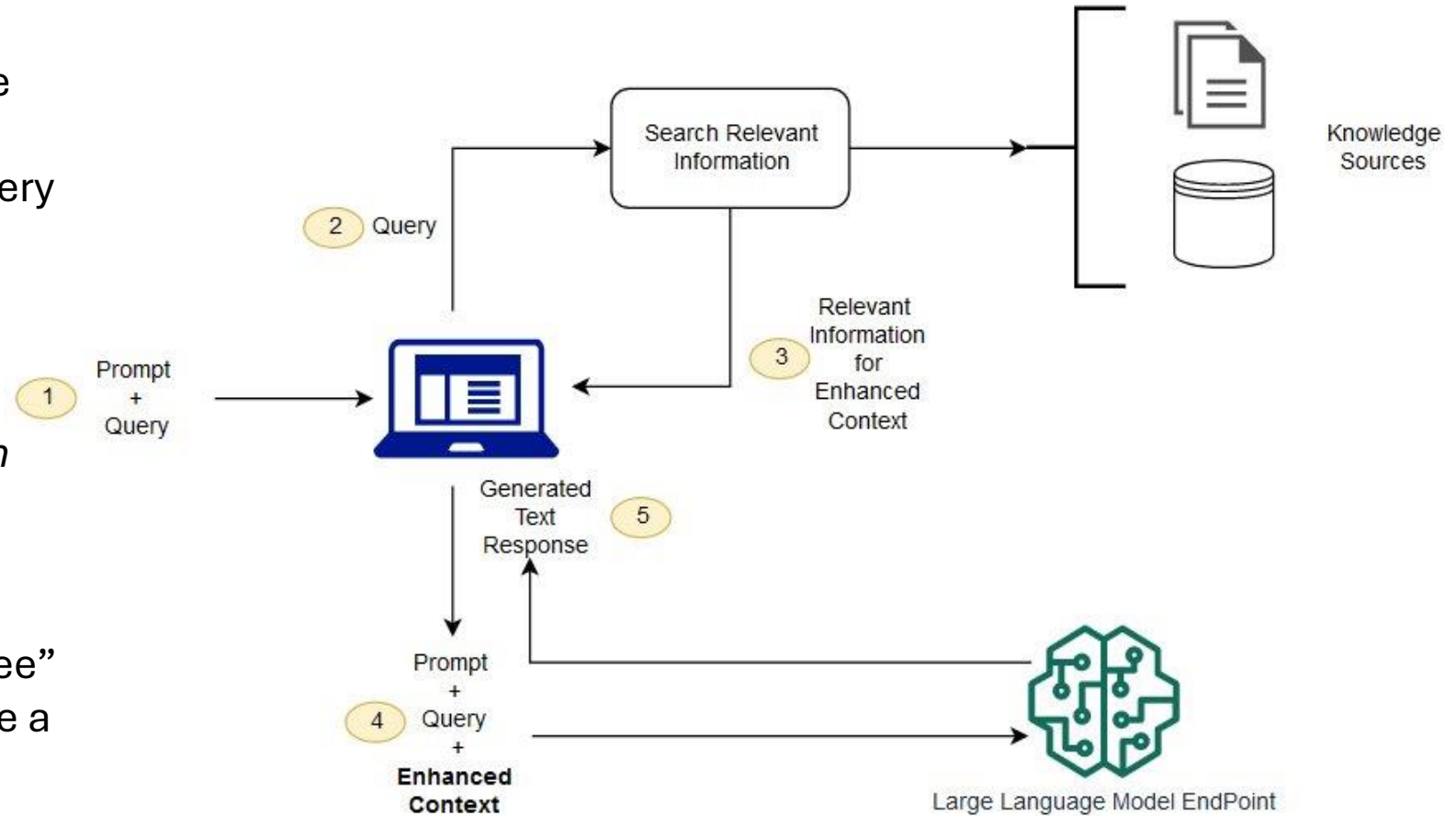
Traditional RAG

- Knowledge source is a Vector Database (traditionally)
- Give LLM instructions to answer the query based on the *relevant information*.

Improvements

Almost all work centers around the *Search Relevant Information* step.

- PDF Extraction (flavor du jour: [docling](#))
- When someone asks “What is the FCCee” you get documents that mention FCCee a lot.
 - Entity extraction can look for definitions of FCCee!
- “What is MATHUSLA” fails because *embedding* doesn’t really grok MATHUSLA.



Entity Extraction

Use an LLM to look for entities in your text as well as relationships!

Define entitles to fit your task!

```
"entity_types": [  
  "physics detector/experiment",  
  "physics concept or theory",  
  "country",  
  "organization",  
  "person",  
  "geo",  
  "event",  
  "category",  
],
```

Given a text document that is potentially relevant to this activity and a list of entity types, identify all entities of those types from the text and all relationships among the identified entities. Use `{language}` as output language.

---Steps---

1. **Identify all entities.** For each identified entity, extract the following information:

- **entity_name:** Name of the entity, use same language as input text. If English, capitalized the name.
- **entity_type:** One of the following types: `[{entity_types}]`
- **entity_description:** Comprehensive description of the entity's attributes and activities

Format each entity as `("entity"{tuple_delimiter}<entity_name>{tuple_delimiter}<entity_type>{tuple_delimiter}<entity_description>)`

2. From the entities identified in step 1, identify all pairs of (source_entity, target_entity) that are *clearly related* to each other.

For each pair of **related entities**, extract the following information:

- **source_entity:** name of the source entity, as identified in step 1
- **target_entity:** name of the target entity, as identified in step 1
- **relationship_description:** explanation as to why you think the source entity and the target entity are related to each other
- **relationship_strength:** a numeric score indicating strength of the relationship between the source entity and target entity
- **relationship_keywords:** one or more high-level key words that summarize the overarching nature of the relationship, focusing on concepts or themes rather than specific details

Format each relationship as

`("relationship"{tuple_delimiter}<source_entity>{tuple_delimiter}<target_entity>{tuple_delimiter}<relationship_description>{tuple_delimiter}<relationship_keywords>{tuple_delimiter}<relationship_strength>)`

LightRAG

Open Source, Popular RAG Framework

- Seems well maintained
- Lots of contributors


The first version I wrote my own RAG pipeline

- This is not difficult! The parts exist in many forms!
- But increasing the response rate/accuracy means adding lots of “after-burners”
- Gets complex fast

LightRAG was the answer I came up with

- By no means unique! I was lead to it by reading comments on reddit!



 LightRAG: Simple and Fast Retrieval-Augmented Generation

Things I really liked about it:

- A WebUI that shows ingestion queue
- Fire and forget ingestion queue
- Containerize-able
- Keyword and entity relationship extraction

I do not use the entity part of the database to its fullest extent!

User Interface

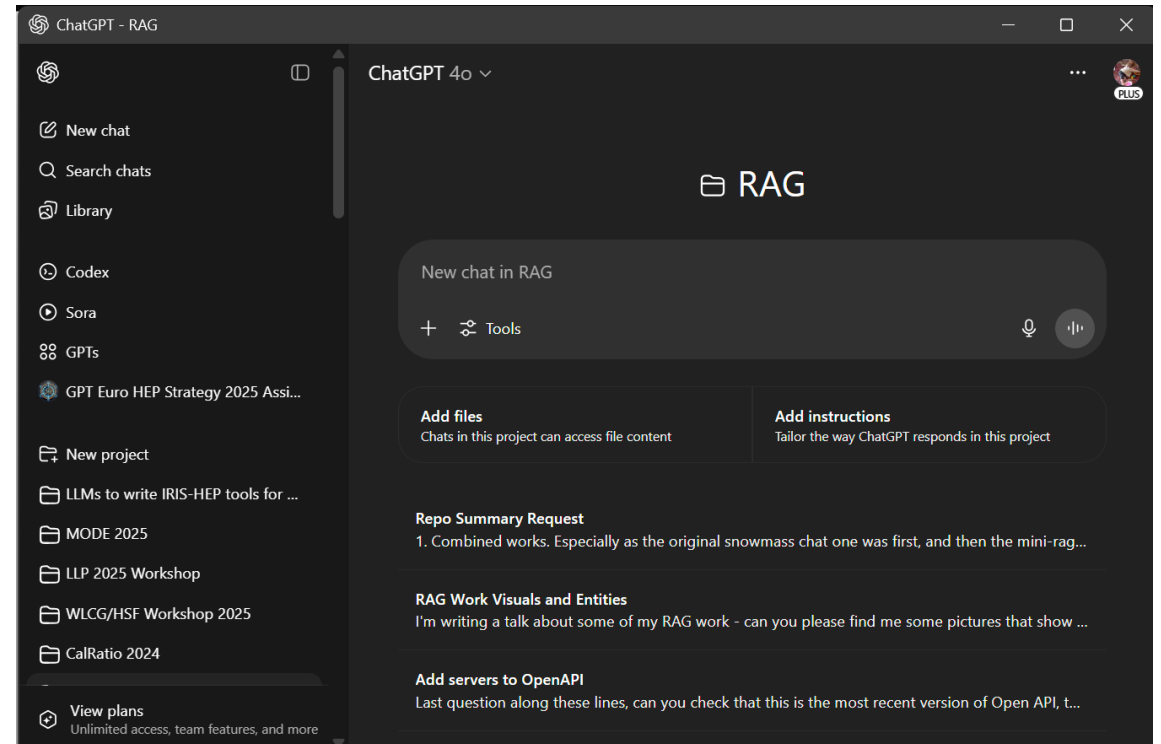
I've been using OpenAI's Plus Subscription

- UI is quite effective
- Built in "voice-mode" is killer for use during commuting
- Collects all my interactions into one (searchable) place

Could this RAG be interfaced?

Can I take advantage of something others much better at programming have built?

- NOT reinvent the wheel! ;-)



Custom ChatGPT's are Possible Answer



You can add a tool

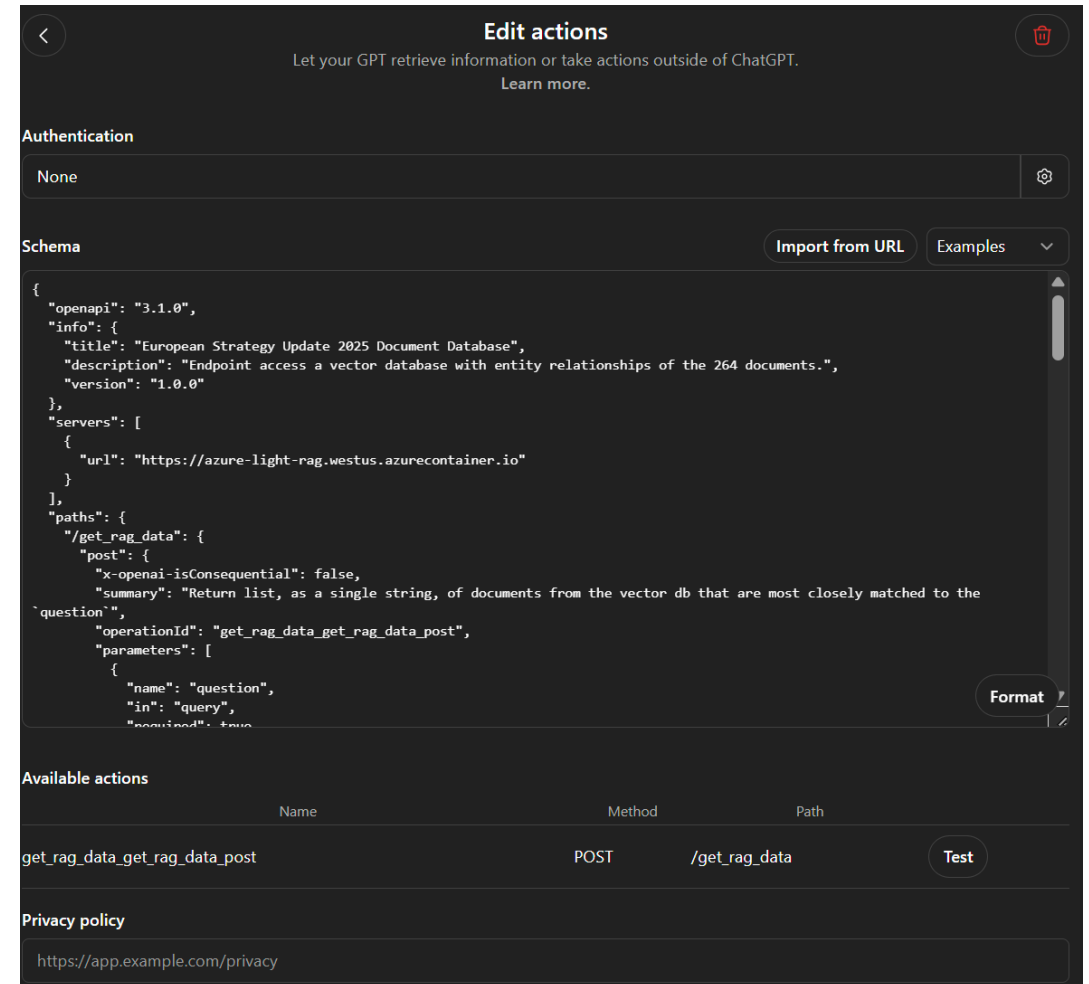
- Uses a REST WebAPI call
- Easy to use

But

- Must be at end of a valid SSL https (with valid cert!)
- Must be publicly visible
- Can't return more than about 7K of text

Learned how to deploy a container with SSL on Azure!

Had to use a LLM to summarize results for a LLM...



Came Close... but needs more work...

- Figure and Table information extraction
 - Docling is quite good at tables, but not clear best way to feed the info to LLM – chunking, etc.
 - As I was preparing this talk noticed Anything-RAG has been integrated with LightRAG...
- Did not figure out how to summarize results properly
 - Too much summarization occurred
 - Lost accurate references
- The Entity Vector Database was huge
 - Had to fit in memory
 - Required a 16 GB container instance – most expensive by far
- Cost of chatting for 3 hours
 - \$1.50 for hosting
 - \$0.50 for LLM usage
- I never solved the copyright/permission issue
 - So this is not public atm

Abstract Ranker

Where do we get the data?

We have a HUGE advantage in HEP

- Years of work to get this – and no one planned for LLM's
- Following the generally good practice of composability.



Many of our **main information websites have API's**

- Any python program can fetch from them easily!
- Popular ones have python libraries
- Easy to build API library yourself if you need it!

We love to hate indico... but...

Conference Ranking



My Interests

```
# Raw prompt for the LLM
abstract_ranking_prompt = """Help me judge the following conference presentation as interesting or
not by summarizing the abstract and ranking it according to topics I'm interested in or not.
"""

interested_topics = [
    "Hidden Sector Physics",
    "Long Lived Particles (Exotics or RPV SUSY)",
    "Analysis techniques and methods and frameworks, columnar analysis, particularly those based around python or "
    "ROOT's DataFrame (RDF)",
    "Machine Learning and AI for particle physics",
    "The ServiceX tool",
    "Distributed computing for analysis (e.g. Dask, Spark, etc)",
    "Data Preservation and FAIR principles",
    "Differentiable Programming",
]

not_interested_topics = [
    "Quantum Computing",
    "Lattice Gauge Theory",
    "Neutrino Physics",
]
```

[Code](#)

This took a fair amount of tuning...
Still not right!

- Doesn't look at authors...

What do I ask for back from the LLM?

```
class AbstractLLMResponse(BaseModel):
    "Result back from LLM grading of an abstract"

    # Summary of the abstract
    summary: str = Field(
        ...,
        title="A short summary of the abstract that does not repeat the title, no more than 200 "
        "characters. If there is no abstract provided, just repeat the title.",
    )

    # The most likely experiment this is associated with
    experiment: str = Field(
        ...,
        title="The Experiment associated with this work if known (ATLAS, CMS, LHCb, "
        "MATHUSLA, etc.). Blank if unknown. No explanation.",
    )

    # List of keywords
    keywords: List[str] = Field(
        ..., title="Short JSON list of string-keywords associated with the abstract"
    )

    # What is the interest level here?
    interest: str = Field(
        ...,
        title="The string 'high', 'medium', or 'low' indicating how interesting I'll find "
        "the abstract.",
    )
```

```
# A short explanation of why the interest level is what it is
explanation: str = Field(
    ...,
    title="Very short explanation of the interest level in the abstract. No more than a "
    "single sentence, 100 words maximum.",
)

# How confident is the AI of its interest assignment?
confidence: float = Field(
    ...,
    title="A float from 0 to 1 representing the confidence in the interest level.",
)

# Any terms in the abstract that the LLM does not know, but would probably make
# the confidence level higher.
unknown_terms: List[str] = Field(
    ...,
    title="Short JSON list of terms (strings) in the abstract whose definition would "
    "improve your confidence.",
)
```

- Use the JSON schema to prompt LLM for formatting
- Does not work for smaller LLM's, but works just right for larger LLM's!

Example: ACAT 2024 – GPT4o-mini

| | | | |
|-------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|----------------------------------------------------------------------------------------------------------------------------|
| "Accelerating Particle Physics Simulations with Machine Learning using Normalizing Flows and Flow Matching" | The talk discusses using Normalizing Flows and Flow Matching in machine learning to enhance particle physics simulations, achieving significant speed-ups and accuracy. | | ['Machine Learning', 'Normalizing Flows', 'Particle Physics Simulation', 'Data Analysis'] |
| Study of columnar data analysis methods to complete an ATLAS analysis | The presentation tests columnar analysis tools, focusing on ServiceX and Coffea, to enhance ATLAS analysis efficiency and publishable results. | ATLAS | ['columnar analysis', 'ServiceX', 'Coffea', 'Run-2 analysis', 'supercomputing'] |
| columnflow: Fully automated analysis through flow of columns over arbitrary, distributed resources | The presentation introduces *columnflow*, a Python toolkit for automating large-scale HEP analyses using distributed resources and various data formats. | | ['automated analysis', 'distributed computing', 'Python', 'machine learning', 'data workflow'] |
| The Good, The Bad, and the Ugly: A Tale of Physics, Software, and ML | The talk discusses using an RNN to improve the identification of long lived particles at the LHC, focusing on software modernization and systematic error control. | ATLAS | ['long lived particles', 'machine learning', 'neural networks', 'software modernization', 'signal discrimination'] |
| ServiceX, the novel data delivery system, for physics analysis | ServiceX addresses data extraction challenges for physics analysis, presenting a python-based system with transformer containers and REST API integration. | | ['ServiceX', 'data delivery', 'python', 'Kubernetes', 'physics analysis'] |
| Quasi interactive analysis of High Energy Physics big data with high throughput | The presentation discusses a new interactive high-throughput data analysis approach for processing large datasets in High Energy Physics, utilizing tools like Dask and ROOT RDataFrame. | | ['High Energy Physics', 'big data', 'data analysis', 'Dask', 'ROOT RDataFrame', 'high-throughput', 'cloud infrastructure'] |
| Optimizing ANN-Based Triggering for BSM events with Knowledge Distillation | This work explores optimizing ANN-based triggering for BSM events using Knowledge Distillation to enhance data processing efficiency at the LHC. | LHC | ['Machine Learning', 'Artificial Neural Networks', 'Knowledge Distillation', 'BSM events', 'trigger optimization'] |

- These are the “most” interesting talks and Posters from ACAT 2024
- Generated with GPT4o-mini
- Cost less than \$0.02 USD

But...

atlas-plot-maker

What about coding?

Goal:

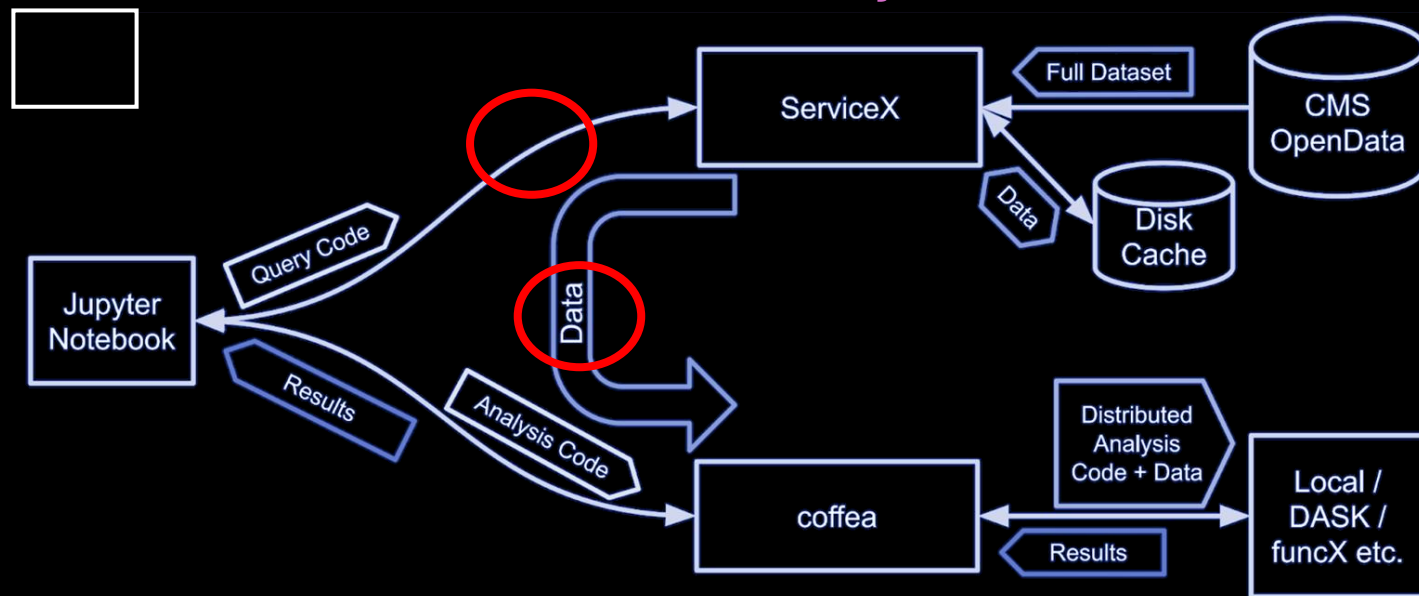
“Plot the jet p_T in MC and Data from 40 to 200 GeV for Run 3”

You get back code and a plot...

Underlying Technology

ServiceX: Data Delivery for HL-LHC Analysis At Scale

Accessible Anywhere



- WebAPI Interface
- Amazon's Object Store API
 - resulting data (S3) in local object store
 - compatible with **python ecosystem** and **ROOT framework**
 - Accessible from laptop to Analysis Facility
 - Support software for local downloads
 - Direct access to data in S3 for AF's

Hardest Problem: writing the code!

Biggest Recent LLM change:

- Very large (tested) context windows!

| Model | Context Window |
|---------------------|-----------------------------------------------|
| GPT-3.5 | 4,096 tokens |
| GPT-4 (legacy) | 8,192 or 32,768 tokens (depending on tier) |
| GPT-4o (4.1) | 128,000 tokens |

You can now pass extensive coding hints to the model!

Prompt caching is going to be key for cost and speed!

Approach:

- Code snippet files with explanation around them
- Files for ServiceX query, awkward array, the Hist package, and the vector package

Tell coding LLM to use instructions...

This approach is well suited to the python ecosystem and Rdata Frame.

It would be interesting to test this out on various large IoC C++ frameworks – like Athena in ATLAS

Testing The Idea

Start with the [Analysis Description Language Benchmarks](#)

1. Plot the ETmiss of all events in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.801167.Py8EG_A14NNPDF23LO_jj_JZ2.deriv.DAOD_PHYSLITE.e8514_e8528_a911_s4114_r15224_r15225_p6697.
2. Plot the pT of all jets in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.801167.Py8EG_A14NNPDF23LO_jj_JZ2.deriv.DAOD_PHYSLITE.e8514_e8528_a911_s4114_r15224_r15225_p6697.
3. Plot the pT of jets with $|\eta| < 1$ in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.801167.Py8EG_A14NNPDF23LO_jj_JZ2.deriv.DAOD_PHYSLITE.e8514_e8528_a911_s4114_r15224_r15225_p6697.
4. Plot the ETmiss of events that have at least two jets with pT > 40 GeV in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.801167.Py8EG_A14NNPDF23LO_jj_JZ2.deriv.DAOD_PHYSLITE.e8514_e8528_a911_s4114_r15224_r15225_p6697.
5. Plot the ETmiss of events that have an opposite-charge muon pair with an invariant mass between 60 and 120 GeV in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.513109.MGPy8EG_Zmumu_FxFx3jHT2bias_SW_CFilterBVeto.deriv.DAOD_PHYSLITE.e8514_e8528_s4162_s4114_r14622_r14663_p6697.
6. For events with at least three jets, plot the pT of the trijet four-momentum that has the invariant mass closest to 172.5 GeV in each event and plot the maximum b-tagging discriminant value among the jets in this trijet in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.601237.PhPy8EG_A14_ttbar_hdamp258p75_allhad.deriv.DAOD_PHYSLITE.e8514_s4369_r16083_p6697.
7. Plot the scalar sum in each event of the pT of jets with pT > 30 GeV that are not within 0.4 in ΔR of any light lepton with pT > 10 GeV.
8. For events with at least three light leptons and a same-flavor opposite-charge light lepton pair, find such a pair that has the invariant mass closest to 91.2 GeV in each event and plot the transverse mass of the system consisting of the missing transverse momentum and the highest-pT light lepton not in this pair.

What Worked

See questions 1-5 [here](#)

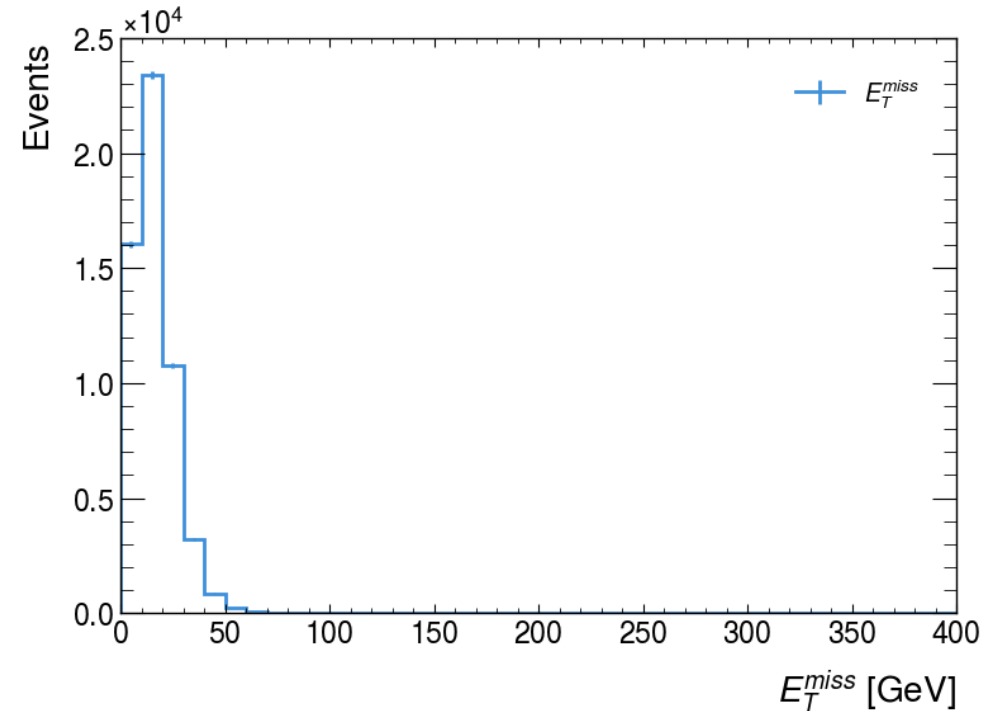
- Used GPT 4.1
- Used copilot's codespaces (free for edu)
 - I suspect this will not remain free

Most complex question that worked:

5. Plot the E_T^{miss} of events that have an opposite-charge muon pair with an invariant mass between 60 and 120 GeV in the rucio dataset

mc23_13p6TeV:mc23_13p6TeV.513109.MGPy8EG_Zmumu_FxFx3jHT2bias_SW_CFilterBVeto.deriv.DA
OD_PHYSLITE.e8514_e8528_s4162_s4114_r14622_r14663_p6697.

See actual code in the [notebook on github](#).



Failure!

6. For events with at least three jets, plot the pT of the trijet four-momentum that has the invariant mass closest to 172.5 GeV in each event and plot the maximum b-tagging discriminant value among the jets in this trijet in the rucio dataset
mc23_13p6TeV:mc23_13p6TeV.601237.PhPy8EG_A14_ttbar_hdamp258p75_allhad.deriv.DAOD_PHYS
LITE.e8514_s4369_r16083_p6697.

See actual failure and comments on produced code in the [notebook on github](#).

What failed:

- Tried to plot a 2D plot with the btagging discriminant on one axis and the invariant mass on the other. Instructions do say “produce a plot and write it out to a png file”...
- Didn’t know how to get the b-tagging discriminant. This is actually quite difficult in ATLAS requiring putting in a “tool” and running it. Need to update hint file
- **The code for slicing the array for the mass closest to 172.5 is not correct**

What do the hint files look like?

All hint files are all in the [hep-programing-hints](#) repo

- When the LLM gets something wrong, add new (isolated) hint till it can perform the task

ServiceX FuncADL xAOD Code Snippets

Use ServiceX to fetch data from `rucio` datasets on the GRID, skimming out only the data required from the events needed.

Fetching data is two steps. First, construct a query. Second, execute the query against a dataset.

A Simple Full Example

This example fetches the jet p_T 's from a PHYSLITE formatted xAOD data sample stored by `rucio`.

```
from func_adl_servicex_xaodr25 import FuncADLQueryPHYSLITE
from servicex_analysis_utils import to_awk
from servicex import deliver, ServiceXSpec, Sample, dataset

# The base query should run against PHYSLITE.
base_query = FuncADLQueryPHYSLITE()

# Query: get all jet pT
jet_pts_query = (base_query
    .SelectMany(lambda evt: evt.Jets())
    .Select(lambda jet: {
        "jet_pt": jet.pt() / 1000.0,
    })
)
```

xAOD Event Data Model Hints

Some hints to help with accessing xAOD objects.

MissingET

Access:

```
query = FuncADLQueryPHYS() \
    .Select(lambda e: e.MissingET().First()) \
    .Select(lambda m: {"met": m.met() / 1000.0'})
```

Despite being only a single missing ET for the event, it is stored as a sequence method from there.

Will add Btagging info here!

Comments On Code

- I used an LLM to generate the first version of the hint files
 - Used ChatGPT's DeepResearch feature
 - Then added as I went. Not much!
 - Clear how crazy our ATLAS Event Data Model can be!
- The fact it got the first 5 questions write in a single shot was amazing
 - One-shot is not the way to go
 - Will want something that runs the code and looks at the results/errors, and iterates
- It is not writing optimal code
 - In the 3-jet example, it reads way too many jets, making the combinatorics way too memory heavy
- We won't work like this
 - First one plot, then add another, etc.
- What is next?
 - Finish out the ADL tests
 - N-shot
 - Can it code the [Analysis Grand Challenge](#) on its own (a ttbar with systematic errors).

Open-Source Models

Open-Source Insanity (the good kind)

How do you test them?

- Getting the right environment installed is very tricky
- Especially using a graphics card
 - Which you have to if you are running locally

I've been through many iterations

- [Docker finally solved this for me](#)
- “it just works” and works on Windows, or Linux from Windows, etc.

But...

- My RTX3080 can't handle more than 7B parameters
- This is barely enough
- For many tasks the big models are much better
 - Starting to see research on using small models
 - For local running on laptop or phone, finally!

Very hard to track

- I've tried the llama, phi, and mistral versions
- RAG turns out to be forgiving
- Saw clear differences in ranking – spending money led to better answers.
 - But I was comparing small models to the large commercial ones, so perhaps not fair!
 - See some comments in backup
- Through put was another issue
 - My poor 3090 serializes the requests.
 - Trying to do entity extraction works, but does not scale. At all.

Was thinking to use a commercial provider for the OSS models

- But then... spending money... May as well...

Closing Remarks

- Web API's make it very easy to try out experiments
- MCP feels like a game changer
 - My 0-shot to 1-shot will likely just be finding the right MCP tool and integrating it
- I'm doing my best to value-add only HEP related items
 - So far I've been able to take advantage of others work
 - Haven't hit a wall
 - Have stayed away from the big frameworks, however!
- Many people have ideas and are trying things out
 - Starting to feel like something more regular should be organized
 - Especially if we can build a small community out of it
- I do not think we've found all the places LLM's will be able to help us push the **physics** of the LHC and other HEP experiments and theory forward.

Backup

Example: ACAT 2024 – GPT4o

| Title (from indico) | Abstract Summary (LLM) | Experiment (LLM) | Keywords (LLM) |
|-------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|------------------------------------------------------------------------------------------------------------------------------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The talk presents a novel inference approach using Ahead-of-time (AOT) compilation for TensorFlow models to reduce memory footprint and improve computational performance in CMS workflows. | CMS | ['Machine Learning', 'TensorFlow', 'Ahead-of-time Compilation', 'High-Energy Physics', 'Data Analysis'] |
| "Accelerating Particle Physics Simulations with Machine Learning using Normalizing Flows and Flow Matching" | Using Normalizing Flows and Flow Matching to accelerate simulations of high-energy physics collision events, achieving faster and accurate predictions compared to traditional methods. | | ['Machine Learning', 'Simulations', 'Normalizing Flows', 'Flow Matching', 'High-Energy Physics'] |
| Modern Machine Learning Tools for Unfolding | Modern machine learning tools are used to perform high-dimensional, unbinned unfolding of LHC data with methods such as event reweighting and direct mapping, accounting for correlations. | LHC | ['machine learning', 'unfolding', 'LHC data', 'event reweighting', 'high-dimensional methods'] |
| Using Legacy ATLAS C++ Calibration Tools in Modern Columnar Analysis Environments | Efforts to adapt legacy ATLAS C++ calibration tools for use in modern columnar analysis environments like Python/Awkward and RDataFrame. | ATLAS | ['ATLAS', 'C++', 'columnar analysis', 'Python', 'RDataFrame', 'calibration tools'] |
| A Function-As-Task Workflow Management Approach with PanDA and iDDS | Overview of the PanDA and iDDS platform for automating multi-step data processing workflows, focusing on python-based user interfaces and distributed task scheduling. | ATLAS | ['Distributed Computing', 'Workflow Management', 'Python', 'Machine Learning'] |
| Fair Universe: HiggsML Uncertainty Challenge | The Fair Universe project is creating an AI ecosystem for HEP to minimize systematic uncertainties and predict confidence intervals using large datasets and advanced analysis techniques. | | ['AI', 'systematic uncertainties', 'confidence intervals', 'Higgs decay', 'tau leptons', 'Codabench', 'NERSC', 'Perlmutter'] |
| AI-driven HPC Workflows Execution with Adaptivity and Asynchronicity in Mind | Investigation into adaptive and asynchronous execution of heterogeneous tasks in scientific workflows using AI-driven middleware for improved resource utilization and reduced costs. | G. Watts (JWASync) | ['AI', 'HPC', 'adaptive execution', 'asynchronous execution', 'middleware'] |

- These are the “most interesting” talks and Posters from the ACAT 2024 workshop.
- Generated with GPT4o
- Cost \$0.43 USD

GPT4o vs GPT4o-mini

GPT-4o

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The talk presents a novel inference approach using Ahead-of-time (AOT) compilation for TensorFlow models to reduce memory footprint and improve computational performance in CMS workflows. | CMS | ['Machine Learning', 'TensorFlow', 'Ahead-of-time Compilation', 'High-Energy Physics', 'Data Analysis'] | 3 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|

GPT-4o-mini

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The presentation discusses a novel inference approach for TensorFlow models in high-energy physics, focusing on AOT compilation to reduce memory usage and improve performance. | CMS | ['Machine Learning', 'TensorFlow', 'AOT Compilation', 'High-Energy Physics', 'Data Analysis'] | 2 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|

Example: ACAT 2024 – Phi3-mini

| | | | |
|--------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------|
| p-adic reconstruction of rational functions in multi-loop calculations | Discusses a new p-adic technique for compact rational function interpolation in multi-loop calculations, requiring fewer samples than traditional methods and resulting in more compact calculations in particle physics. Not quantum computing, lattice theory, or neutrino-specific. 'high' The talk directly addresses techniques using p-adic numbers to improve the efficiency and compactness of multi-loop scattering amplitude calculations, which is within the scope of my interests, especially concerning computational methods applicable to theoretical physics. | Blank | ['p-adic reconstruction', 'multi-loop calculations', 'computer algebra', 'interpolation', 'rational functions', 'particle physics'] |
| Optimizing the ATLAS Geant4 detector simulation | The ATLAS experiment enhanced Geant4 simulation for better FL-LHC throughput with recent Woodcock tracking integration and various optimizations like software enhancements, simplifications in geometry/magnetic field descriptions, and code-Geant4 interface tweaks leading to over 100% improvement in performance. | ATLAS | ['Geant4', 'ATLAS', 'Monte Carlo Simulation', 'Performance Optimization', 'Electromagnetic Calorimeter', 'Run 3', 'HL-LHC'] |
| Fast and Precise Amplitude Surrogates with Bayesian and Symmetry Preserving Networks | Deep learning with symmetries for accurate matrix elements scaling despite multiple particles in interactions | | ['Deep learning', 'Symmetry preservation', 'Bayesian networks', 'Machine learning'] |

- Phi-3 model is much more chatty!
- Has VERY different answers
- No real synthesis for summary
- ~3B parameter model
- Took about 15 minutes to run on a A100, 45 on a RTX 3080

GPT4o vs GPT4o-mini vs Phi3.5-Mini

GPT-4o

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The talk presents a novel inference approach using Ahead-of-time (AOT) compilation for TensorFlow models to reduce memory footprint and improve computational performance in CMS workflows. | CMS | ['Machine Learning', 'TensorFlow', 'Ahead-of-time Compilation', 'High-Energy Physics', 'Data Analysis'] | 3 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|

GPT-4o-mini

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The presentation discusses a novel inference approach for TensorFlow models in high-energy physics, focusing on AOT compilation to reduce memory usage and improve performance. | CMS | ['Machine Learning', 'TensorFlow', 'AOT Compilation', 'High-Energy Physics', 'Data Analysis'] | 2 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|

Phi3.5-Mini

No real synthesis

| | | | | | |
|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The TensorFlow AOT model compilation talk discusses deploying machine learning models in particle physics with low memory and higher computational efficiency for better real-world production scenarios. Integration practices and strategies are also presented for practical implementation in physics experiments. A focus on performance improvement for central IT infrastructure utilization in high-energy physics experiments. | CMS | ['Machine Learning', 'Distributed Computing', 'Data Preservation', 'AOT Compilation', 'TensorFlow models', 'Deployment', 'High Energy Physics', 'Resource Optimization'] | 2 | Poster |
|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|--------|

Example: ACAT 2024 – Phi3-small

| | | | |
|-------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Optimizing the ATLAS Geant4 detector simulation | Enhancements to Geant4 simulation for ATLAS experiment, improving efficiency by 100%+ since Run 2 | ATLAS | ['Geant4', 'detector simulation', 'Electromagnetic Calorimeter', 'Woodcock tracking', 'Optimization'] |
| "Accelerating Particle Physics Simulations with Machine Learning using Normalizing Flows and Flow Matching" | Utilizes machine learning, specifically Normalizing Flows, to speed up and improve accuracy of high-energy physics simulations. Discusses oversampling for uncertainty reduction. Short list of keywords associated with the abstract: ['Machine Learning', 'Normalizing Flows', 'Simulation', 'HEP']. 'high': Given my interest in Machine Learning and AI for particle physics, this talk is highly interesting. It pertains directly to the development of analysis frameworks using ML, which aligns well with my interest in improving current methodologies for HEP simulation analysis and meets several of the other topics I am keen on such as Data Analysis techniques and Distributed computing for analysis, though the talk doesn't directly address distributed computing approaches or FAIR data principles. Nonetheless, the proposed approach for direct generation of final data could potentially make use of these topics. | Not specified | ['Machine Learning', 'Normalizing Flows', 'Simulation', 'HEP'] |
| Modern Machine Learning Tools for Unfolding | Modern ML tools applied to LHC data unfolding, comparing known/unveiled methods for performance control in many dimensions. | LHC | ['unfolding', 'machine learning', 'LHC', 'data analysis', 'performance control'] |
| Using Legacy ATLAS C++ Calibration Tools in Modern Columnar Analysis Environments | Adapting legacy C++ calibration tools for ATLAS experiments to fit columnar analysis in modern environments with minimal code changes for on-the-fly calculations in Python Jupyter notebooks G. Watts (UW/Seattle) | ATLAS | ['ATLAS', 'Legacy C++ Tools', 'Columnar Analysis', 'Compatibility Challenges', 'Minimal Code Modifications', 'On-the-fly Calibration', 'Python Jupyter Notebook'] |

- Better, but not by a lot
- ~8B parameter model
- Took about 45 minutes to run on a A100. Could not fit on a 3080.

GPT4o vs GPT4o-mini vs Phi3.5-Mini

GPT-4o

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The talk presents a novel inference approach using Ahead-of-time (AOT) compilation for TensorFlow models to reduce memory footprint and improve computational performance in CMS workflows. | CMS | ['Machine Learning', 'TensorFlow', 'Ahead-of-time Compilation', 'High-Energy Physics', 'Data Analysis'] | 3 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|---|--------|

GPT-4o-mini

| | | | | | |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | The presentation discusses a novel inference approach for TensorFlow models in high-energy physics, focusing on AOT compilation to reduce memory usage and improve performance. | CMS | ['Machine Learning', 'TensorFlow', 'AOT Compilation', 'High-Energy Physics', 'Data Analysis'] | 2 | Poster |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------|---|--------|

Phi3.5-Mini

| | | | | | |
|---------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------|---|--------|
| Ahead-of-time (AOT) compilation of Tensorflow models for deployment | Talk on using AOT compilation to improve memory usage and performance of TensorFlow models for deployment in particle physics applications | CMS | ['TensorFlow', 'AOT compilation', 'Particle physics', 'Machine learning', 'CMS experiment'] | 2 | Poster |
|---------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------|---|--------|