



# An AI assistant for the ATLAS collaboration

Daniele Dal Santo<sup>1</sup>, Juerg Beringer<sup>2</sup>, [Joe Egan](#)<sup>3</sup>, Ben Elliot<sup>3</sup>, Gabriel Facini<sup>3</sup>, Daniel Murnane<sup>2,4</sup>, Samuel Van Stroud<sup>3</sup>, Alex Sopio<sup>5</sup>, Jeremy Couthures<sup>6</sup>, Joe George<sup>3</sup>, Runze Li<sup>7</sup>, Cary Randazzo<sup>8</sup>

IML Working Group, 1st July 2025

(1)



(2)



BERKELEY LAB

(3)



(4)



UNIVERSITY OF  
COPENHAGEN

(5)



(6)



(7)



(8)



# Contents

---

- The problem we're trying to solve
- What we've done so far
- Where we're going

# The Problem: Search

---

- CERN experiments have a lot of documentation, spread over myriad forms (twiki, CDS, Indico, Gitlab mkdocs...)
- From the [ATLAS Big Survey](#) (slides)

General satisfaction with zoom and Indico, Glance and in particular CDS with room for improvement.

**Searchability** is a common **weakness** of all tools.  
~400 comments received with following key suggestions:

→ **Glance**: API, speed, search User Interface (UI)  
→ **CDS**: search, document review, editor, UI ...

- Navigating documentation is something *everyone has to do*
  - Making this more straightforward can **benefit everyone**

# The Problem: LLMs

---

- No Big Survey question on LLM usage
  - Anecdotally: It is extremely prevalent, especially for coding
- LLMs aren't trained on internal documents (so they say)
- Demand for an ATLAS-specific tool is there

# The Problem: LLMs

---

- Some quotes from our own survey (more on this later)

*"It would help in many ways... helping to find all the relevant information in ATLAS when needed."*

*"I would like to start using it as a helper, a copilot for example for evaluating code."*

*"Coding and text reviewing for sure... but also finding information that search might not easily return."*

*"The functionality that you're adding with the search and with the chat, I would call quite valuable."*

*"Since it's a collaboration, I would be more open to leave feedback than for commercial tools... because it's my collaboration."*

*"If it's something within our collaboration, then I would be more confident to add an Athena piece of code."*

*"If [a specialized agent] gets me something more than just filtered answers—like drafts of configuration files—that would be very useful."*

*"Probably the Athena Code Assistant is the most important for my work."*

*"I was not expecting all these assistant levels... The Data Analysis Mentor is interesting. I'd be very interested to try that, though it's probably the most difficult one to implement."*

*"I think the feedback system with the rating and the number of users gives you confidence. A high score from many users builds trust."*

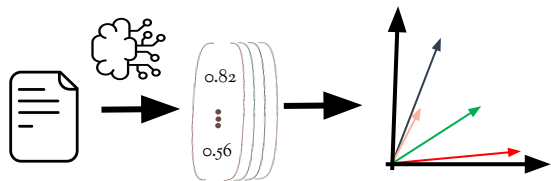
*"What I would like is to point to the GitLab repo and then say—I'm thinking of introducing this code snippet—will it break anything?"*

*"If there's one that could help me find where a function is implemented in the Atlas software, or where it's used—then that's a big bonus."*

# Solution to both? RAG

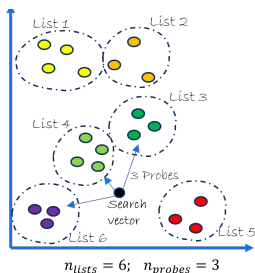
~Unified search engine for internal documents across different sources

## 01 Embeddings



**Transformer** model converts **paragraphs** into **vectors** encoding semantic meaning.

## 02 Retrieval



A **Cosine** Similarity Search via **Approximate** Nearest Neighbors finding the closest matching vectors.

## 03 Generation



**LLM** uses retrieved documents to **answer** the question asked

# Where we are

# Current Status (1st July 2025)

---

Sources

[chATLAS\\_Scrape](#)

Embeddings

[chATLAS\\_Embed](#)

RAG

[chATLAS\\_Chains](#)

Benchmark

[chATLAS\\_Benchmark](#)

Frontend

- **Health warning:** Not mandating what best practice is, just sharing what we have done
  - Discussions on other approaches always welcome!
- Backend code split into four creatively named packages
  - Uses industry standard libraries (LangChain)
  - Open source (without internal data)
  - Modular, reusable
  - *Not ATLAS-specific*



# Current Status: Sources

Sources

Embeddings

RAG

Benchmark

Frontend

- 5 sources available

| Name          | Acquisition Method        | # Parent chunks | # Child Chunks |
|---------------|---------------------------|-----------------|----------------|
| Twiki         | Direct access via CERN IT | 148,707         | 661,153        |
| CDS           | Scraping                  | 224,965         | 1,029,869      |
| Indico        | Scraping                  | 6,491           | 32,969         |
| ATLAS Talk    | API                       | 29,294          | 233,162        |
| Gitlab mkdocs | API (GraphQL)             | 45,838          | 201,681        |

# Current Status: Embeddings

Sources

Embeddings

RAG

Benchmark

Frontend

## Embedding Creation

- Embedding models map text to vectors
- [Finetune model](#) on Q&A pairs from ATLAS Talk
  - < 1 hour on 40 GB A100

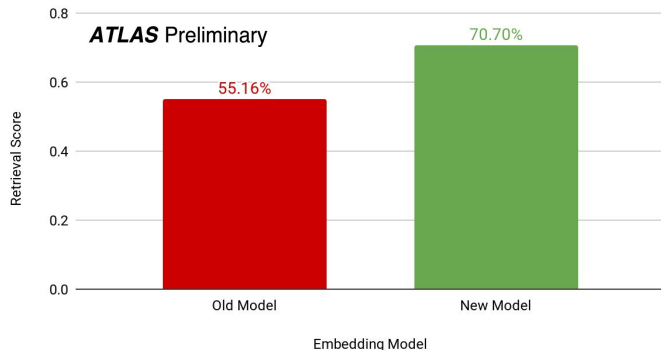
**N.B.** *Not a finetuned LLM*

(resource restrictions)

## Embedding Storage

- Migrated vectorstore from ChromaDB to *postgreSQL*
  - Removes need for storing entire DB in memory
- Hosted at CERN via [Database On Demand](#) service
  - Stable and *scalable* as tool grows in popularity

Retrieval Score vs. Embedding Model



# Current Status: RAG

---

Sources

Embeddings

**RAG**

Benchmark

Frontend

- Deployed: Vanilla parent-child RAG chain
  - GPT 4o mini for generation
  - More expensive models available with own API key
- **New\***: Support for Groq-hosted models
  - Significantly lower latency for API calls
  - Crucial for developing agentic systems

*\*access from Florian and AccGPT team, thanks!*

# Current Status: Benchmark

Sources

Embeddings

RAG

**Benchmark**

Frontend

- *Goal:* Move from “vibe-based” to more objective evaluation
- Four metrics to evaluate performance

|                                    |  |
|------------------------------------|--|
| <a href="#">AnswerCorrect</a>      | Test for evaluating correctness of the answer as judged by an LLM  |
| <a href="#">DocumentMatch</a>      | A test to see whether the returned documents from the RAG match the document the question was generated on Has a base score which just determines whether the correct document was returned at all as well as a weighted score which weights earlier returned documents as better. |
| <a href="#">LexicalMetrics</a>     | Tests to carry out exact match, F1 and rouge score on the answers from the RAG compared to ground truth answers.   |
| <a href="#">semanticSimilarity</a> | A script to perform a semantic similarity test on the RAGs answer compared to the benchmark answer.  |

- Question set: 150 questions from top viewed twikis
- Quality control:
  - **Multiple personas** prompt for diverse questions
  - Manual screening

# Current Status: Frontend

Sources

Embeddings

RAG

Benchmark

Frontend

- Hosted using CERN's [Platform-as-a-Service](#)
- [Main](#) app for production, [staging](#) for newest updates
- UI written from scratch, *clunky but functional*
  - Looking to switch to external library, e.g. [OpenWebUI](#)



# Current Status: Frontend

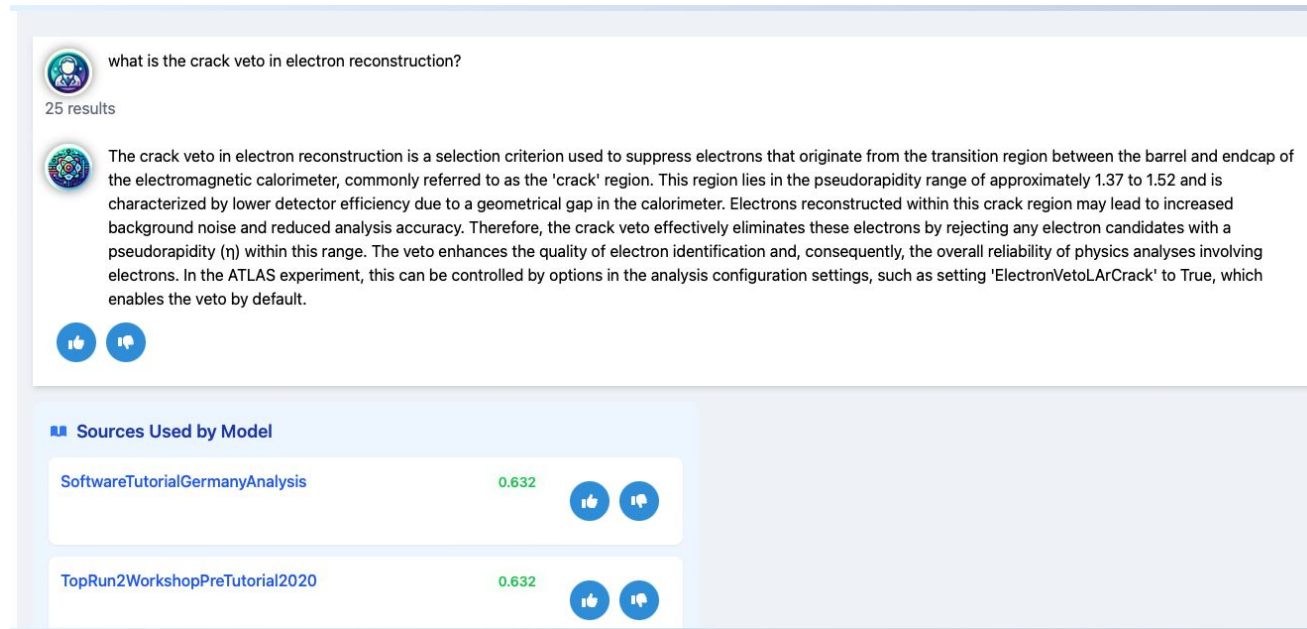
Sources

Embeddings

RAG

Benchmark

Frontend



The screenshot displays a user interface for a Retrieval-Augmented Generation (RAG) system. At the top, a search bar contains the query "what is the crack veto in electron reconstruction?". Below the search bar, it indicates "25 results". A search result is shown with a circular icon and a text snippet. The text snippet describes the crack veto in electron reconstruction, mentioning the transition region between the barrel and endcap of the electromagnetic calorimeter, the 'crack' region, and the selection criterion used to suppress electrons. Below the text snippet, there are two circular icons: a thumbs up and a thumbs down. Below the search results, there is a section titled "Sources Used by Model". This section lists two sources: "SoftwareTutorialGermanyAnalysis" and "TopRun2WorkshopPreTutorial2020". Each source has a score of 0.632 and two circular icons: a thumbs up and a thumbs down.

what is the crack veto in electron reconstruction?

25 results

The crack veto in electron reconstruction is a selection criterion used to suppress electrons that originate from the transition region between the barrel and endcap of the electromagnetic calorimeter, commonly referred to as the 'crack' region. This region lies in the pseudorapidity range of approximately 1.37 to 1.52 and is characterized by lower detector efficiency due to a geometrical gap in the calorimeter. Electrons reconstructed within this crack region may lead to increased background noise and reduced analysis accuracy. Therefore, the crack veto effectively eliminates these electrons by rejecting any electron candidates with a pseudorapidity ( $\eta$ ) within this range. The veto enhances the quality of electron identification and, consequently, the overall reliability of physics analyses involving electrons. In the ATLAS experiment, this can be controlled by options in the analysis configuration settings, such as setting 'ElectronVetoLArCrack' to True, which enables the veto by default.

Sources Used by Model

| Source                          | Score | Feedback       |
|---------------------------------|-------|----------------|
| SoftwareTutorialGermanyAnalysis | 0.632 | Thumbs Up/Down |
| TopRun2WorkshopPreTutorial2020  | 0.632 | Thumbs Up/Down |

# Work In Progress

# Sources: Gitlab Mkdocs

Credit: Ben Elliot & Sam van Stroud (UCL)

- Automated scrape/update pipeline using [Gitlab CI/CD runner](#)
  - Built on GraphQL
- Weekly execution, only scrape projects updated that week
- Filter out:
  - Projects containing blacklisted patterns
  - Projects with fewer than 5 markdown files, *not counting*:
    - CHANGELOG files
    - Files with fewer than 100 or more than 100k characters
  - Forks
  - Personal projects
  - Empty repos
  - Inactive projects (no commit in last 120 days)
  - Pull mirrors



# Sources: Twiki

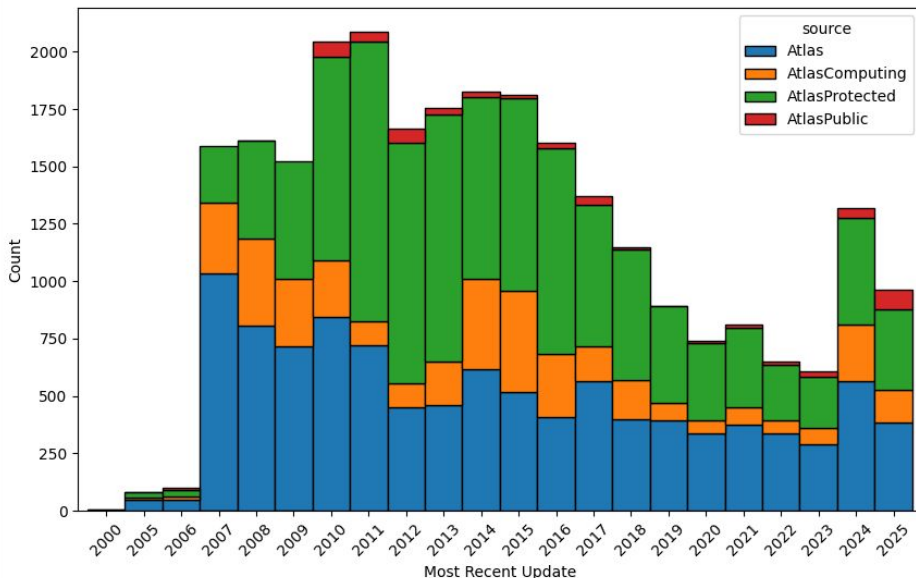
Credit: Joe Egan, Ben Elliot & Gabriel Facini (UCL)

- ~16k Twikis after some basic selection
- Of *varying quality*

## New Approach

Thanks to James Catmore for the suggestion

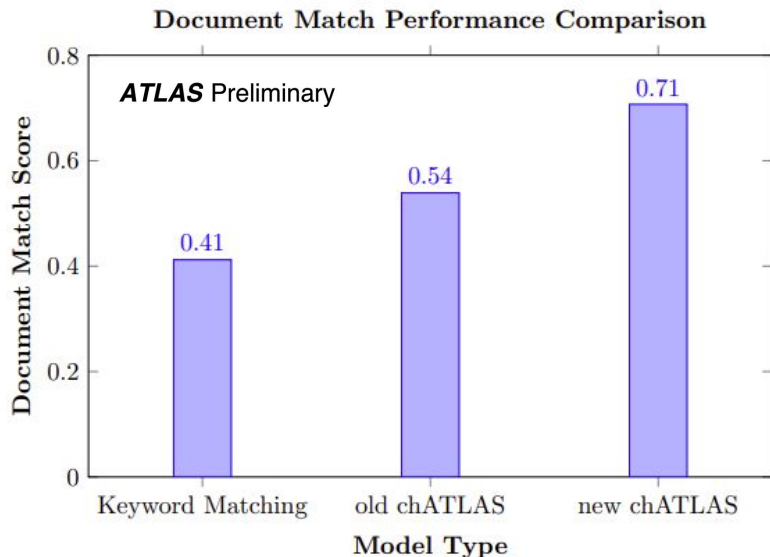
- Define top level twikis
- Find links on each page, add to level below
  - Only links to other twikis
- Continue for 3 levels
- Results in 6k *important twikis*
  - Using these to create an improved benchmark (more later)
  - Plan is to use for frontend DB



# Embedding: Architecture Optimisation

Credit: Juerg Beringer (Berkeley)  
& Ben Elliot (UCL)

- Computing similarity scores is expensive
- Support for different vector DB search algorithms
  - IVFFlat (default)
  - HNSW
  - Exact search
- Each has tradeoffs w.r.t recall, speed, index build time, size
- PGvector default parameters for IVFFlat and HNSW not optimal
  - Tuning required



# Benchmark: Result Visualisation

Credit: Joe Egan, Ben Elliot (UCL)

- Goal: Sandbox for efficiently testing different RAG chains
- First version running, lots of room for improvement

## Compare Single Component

Variable to Compare:

llm\_name

Llm Name Comparison

Fix Other Variables:

Search Type:

All

K:

All

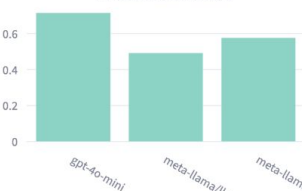
K Text:

All

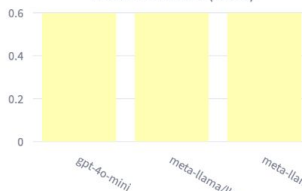
Model Name:

All

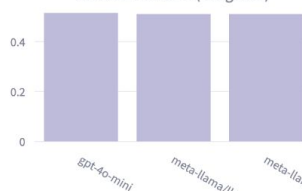
Answer Correctness



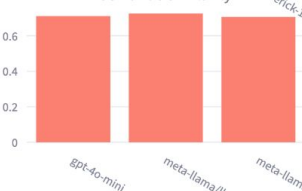
Document Match (Mean)



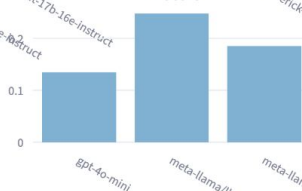
Document Match (Weighted)



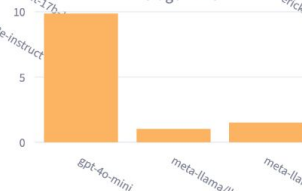
Semantic Similarity



F1 Score



Average Time

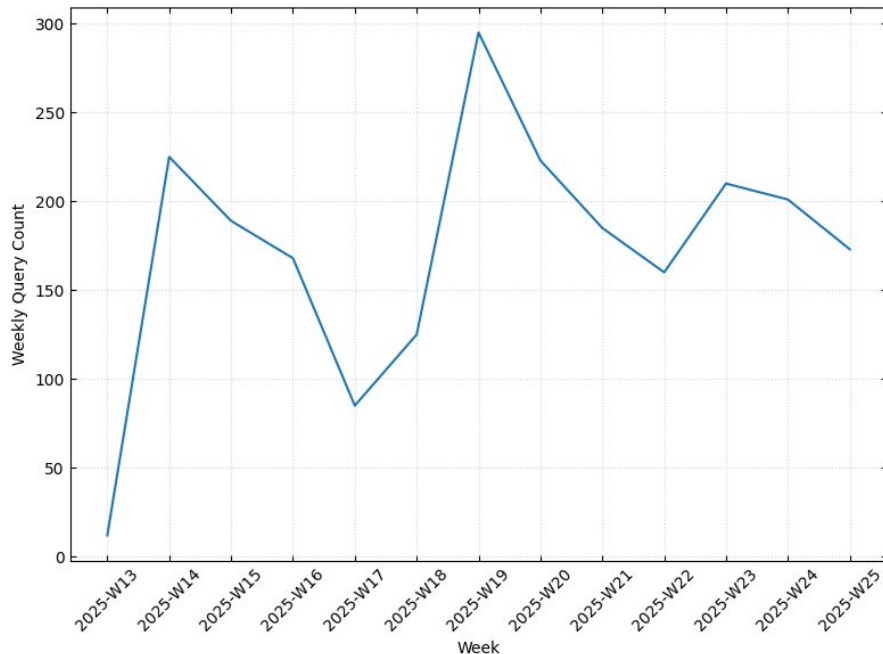


# Benchmark: Improving robustness

Credit: Joe Egan, Ben Elliot &  
Gabriel Facini (UCL)

## User questions

- Logging query-answer pairs for ~3 months, eventual plan is:
  - Responses with positive feedback go into benchmark
  - Responses with negative feedback require manual review
- Currently not enough responses with feedback for this to be useful
  - Supplement benchmark with manually reviewed responses



# Benchmark: Improving robustness

---

*Credit: Joe Egan, Ben Elliot &  
Gabriel Facini (UCL)*

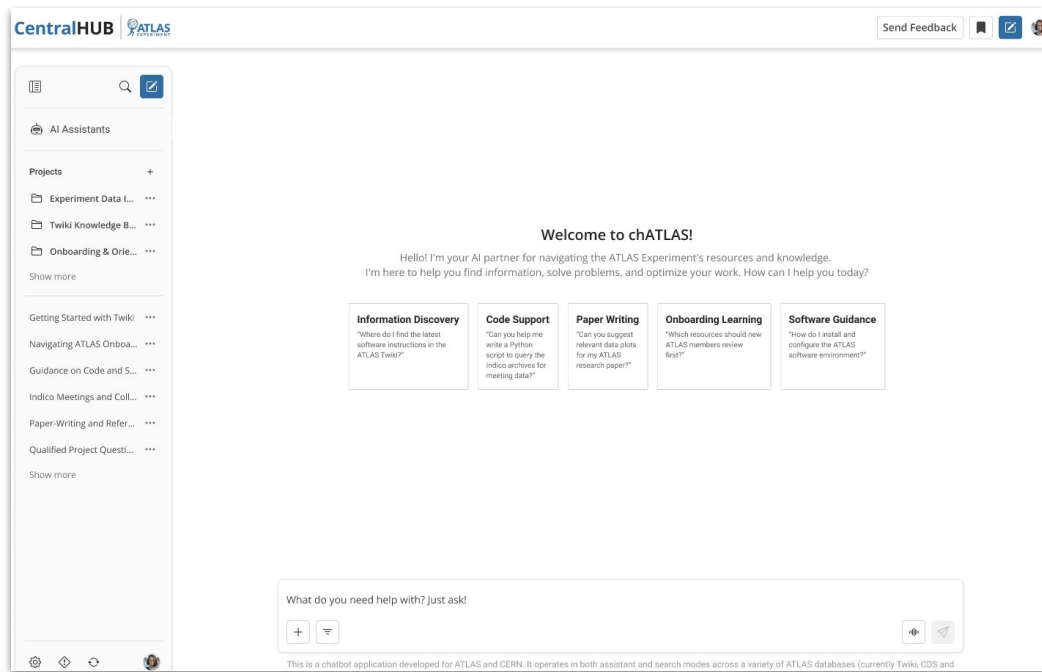
## LLM generated questions

- Increase benchmark coverage with less personpower
  - 750 important twikis linked from the top level ones
  - Several validation stages using an LLM (see right)
  - Questions passing these are reviewed manually
1. Relevant to ATLAS?
  2. Outdated?
  3. Appropriate technical depth?
  4. Completely answerable from document?
  5. Answer is factually accurate?
  6. Answer represents the document content?

# Frontend

Credit: Joan Toribio Agualeles, Maria Francisca Pinto Pinheiro Torres & Radina Milenova Marinova (University of Aalborg)

- Ongoing project with Aalborg University design students
- Survey, user interviews and beta testing
- **85%** think ATLAS knowledge base is overwhelming
- **72%** highlighted citation as the most important feature in an AI assistant
- Cleaner design out-of-the-box with [Open WebUI](#)



# Summary

- [chATLAS](#) built as an information finding tool
- Growing user base brings new challenges
  - Scalability and reliability
  - Professional looking UI
- Developed a modular, open source codebase
  - Want maintainability as the project grows
- More complex RAG  $\neq$  improved performance
  - Focus on building a **robust, comprehensive benchmark**

