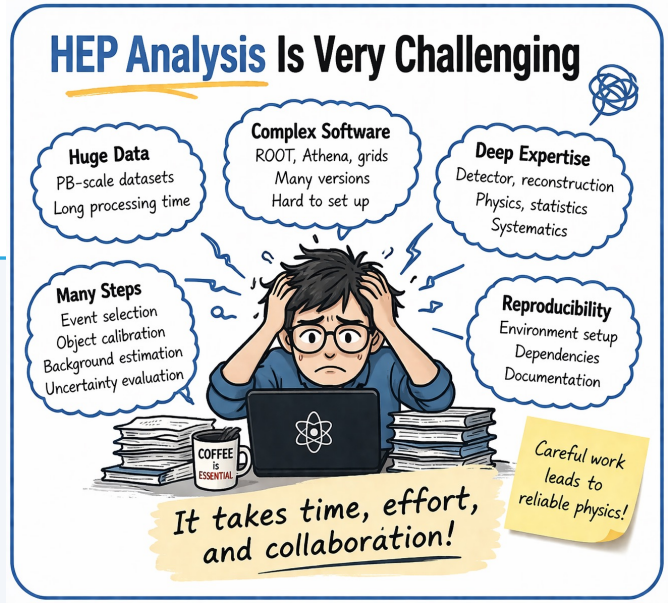
Reproducing a paper result using LLMs

Published on arXiv:2604.14696

FJPPN — Japan-France Workshop on Computing Technologies at KEK
Tomoe Kishimoto*, Masahiko Saito**
*Computing Research Center, KEK
** International Center for Elementary Particle Physics, The University of Tokyo

加速器だから見える世界。
KEK

Introduction and Background



Open access & reproducibility

- Open access is growing, and more studies publish data alongside papers
- This promotes scientific progress and societal impact
- It also strengthens the reproducibility of published results
- If the paper is well-described and the data is public, the result should be reproducible

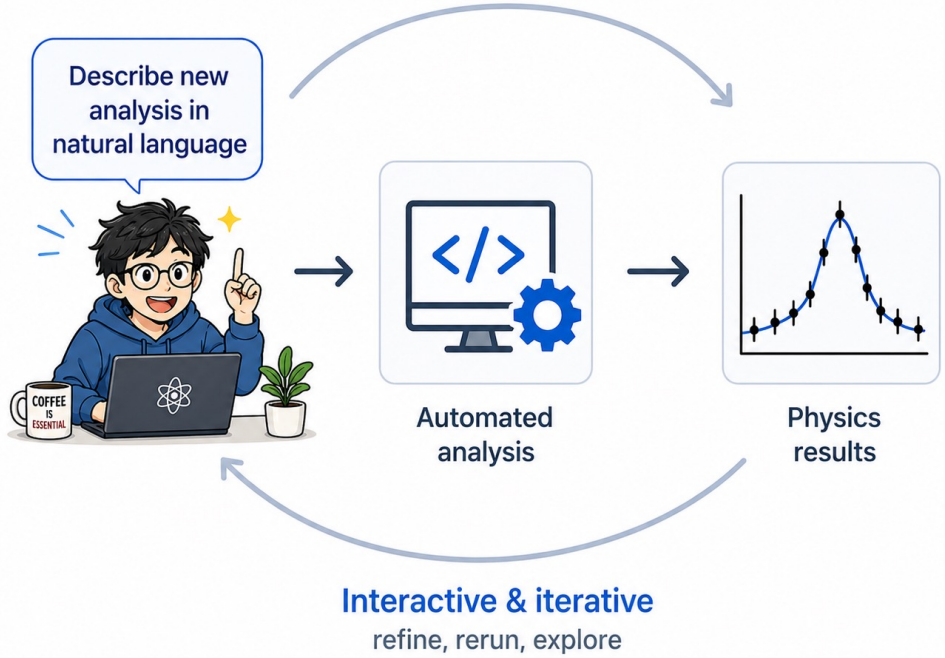
But... HEP has a unique challenge

- Large HEP experiments rely on dedicated analysis software (ROOT, Athena…) and computing environment
- Understanding these stacks requires substantial domain expertise
- Even when data is public, reproducing an analysis is hard for outsiders

Research Goal

Goal

Build a framework that automatically generates an analysis environment and code from HEP publications, and verifies the reproducibility of the results.



Reliability & feedback

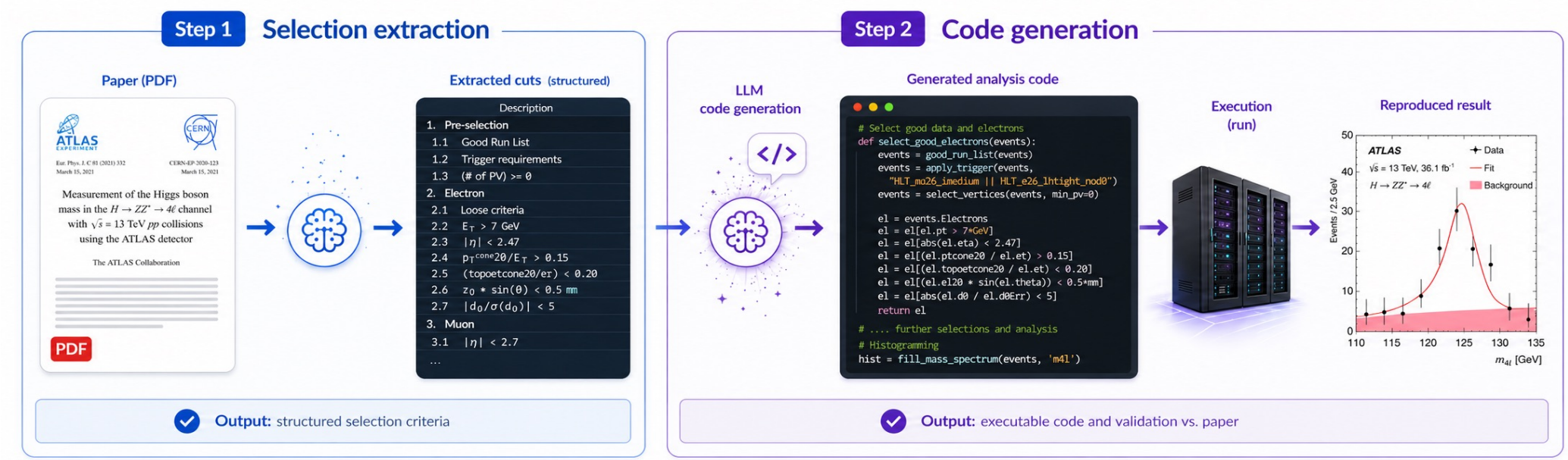
- Improve the reliability of published results
- Provide constructive feedback to authors (missing / ambiguous descriptions)

Toward analysis automation

- **Once paper → code is established, users could describe new analyses in natural language**
- Long-term vision: interactive, automated HEP analysis workflows

Project Overview: PoC for Reproducibility

Two-stage LLM pipeline: paper → selection criteria → analysis code → result



- **Step 1 — Selection extraction**: LLMs read the paper (and references) and produce a structured list of selection criteria
- **Step 2 — Code generation**: LLMs turn the criteria into executable analysis code, which is run and compared with the paper
- Each step is evaluated separately so we can see where current LLMs succeed and where they fail

Benchmark: ATLAS H → ZZ* → 4ℓ analysis

Target analysis

- ATLAS H → ZZ* → 4ℓ (arXiv:1806.00242)
- Data: ATLAS Open Data, pp collisions (2015–16)
- Why this paper?
  - Physics significance (Higgs boson)
  - Should be publicly reproducible via Open Data
  - Non-trivial: selection details are delegated to references

Ground truth (manually curated)

- We manually reproduced the analysis (baseline code)
- **27 selection cuts** extracted from the paper and its references
  - Only cuts implementable via ATLAS Open Data are included

H→ZZ→4l

- 🟢 → 文章に記載がある。再現可能。
- `Partial` → 文章に記載があるが、詳細な実装手順が不明。他の論文などを参考にすれば再現可能か
- `NG` → 文章に記載なし。再現不可。

Description	Main	Ref.[11]	Ref. [44]	Details
1. Pre-selection				
1.1 `Good Run List`	`Partial`	`Partial`	`NG`	(1)
1.2 `Trigger requirements`	`Partial`	`Partial`	`Partial`	(2)
1.3 `(# of PV) > 0`	`NG`	🟢	🟢	(3)
2. Electron				
2.1 `loose criteria`	`NG`	🟢	🟢	(4)
2.2 `eT > 7GeV`	🟢	🟢	🟢	
2.3 `abs(eta) < 2.47`	🟢	🟢	🟢	
2.4 `(ptcone20/eT) < 0.15`	`Partial`	🟢	🟢	(5)
2.5 `(topoetcone20/eT) < 0.20`	`Partial`	🟢	🟢	(5)
2.6 `z0 * sin(theta) < 0.5mm`	`NG`	🟢	`NG`	(6)
2.7 `d0/sig(d0) < 5`	`NG`	`NG`	🟢	(6)
3. Muon				
3.1 `abs(eta) < 2.7`	🟢	🟢	🟢	
3.1.1 `type == 2,3 for abs(eta) < 0.1`	`NG`	🟢	🟢	(7)
3.1.2 `type == 0 for 0.1 < abs(eta) < 2.5`	`NG`	🟢	🟢	(7)
3.1.3 `type == 1 for 2.4 < abs(eta) < 2.7`	`NG`	🟢	🟢	(7)
3.2 `pT > 5 GeV`	🟢	🟢	🟢	
3.2.1 `pT > 15 GeV for type == 3`	`NG`	`NG`	🟢	
3.3 `(ptcone30/pT) < 0.15`	`Partial`	🟢	🟢	(5)
3.4 `(topoetcone20/pT) < 0.30`	`Partial`	🟢	🟢	(5)
3.5 `z0 * sin(theta) < 0.5mm`	`NG`	🟢	`NG`	(6)
3.6 `d0/sig(d0) < 3`	`NG`	`NG`	🟢	(6)
3.6 `abs(d0) < 1`	`NG`	🟢	`NG`	(6)
4. Quadruplet				
4.1 `(# of SFOS pairs) >= 2`	🟢	🟢	🟢	
4.2 `50 < m12 < 106 GeV`	`NG`	🟢	🟢	
4.3 `12 < m34 < 115 GeV`	`NG`	🟢	🟢	
4.4 `pT > 20, 15, 10 GeV`	🟢	🟢	🟢	
4.5 `dR(l,l) > 0.1 (0.2) for SF (OF)`	`NG`	🟢	🟢	
4.6 `mll > 5 GeV for SFOS leptons`	`NG`	🟢	🟢	
4.7 `4 leptons vertex fit`	`Partial`	`Partial`	`Partial`	(8)
4.8 `(# of CB) >= 3 for mm`	`NG`	🟢	🟢	
4.9 `Z mass constraint`	`Partial`	`NG`	`Partial`	
4.10 `FSR correction`	`Partial`	`Partial`	`Partial`	
4.11 `110 < m < 135 GeV`	🟢	🟢	🟢	

- (1) コード実装にはxmlファイルの詳細が必要
- (2) Non prescaled trigger。コード実装にはTriggerの詳細が必要
- (3) At least two tracks with pT > 400 MeV
- (4) DFCommonElectronsLHLooseBL
- (5) CloseByCorrの補正が必要
- (6) PVに対する距離に補正が必要
- (7) 0: Combined, 1: MS standalone, 2 Segment tagged, 3 Calo
- (8) FourLeptonVerticesAux

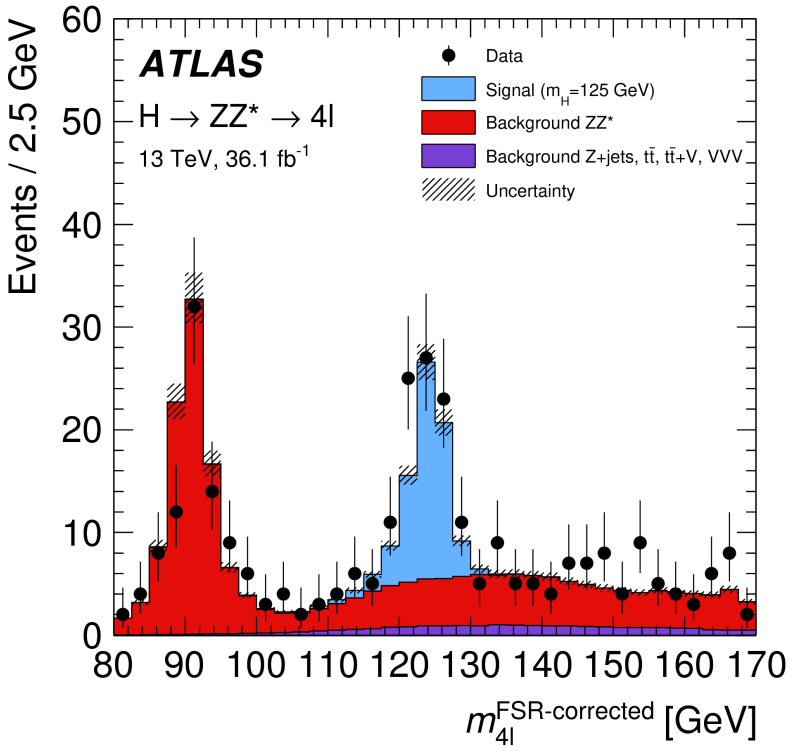
Some of the cuts are described only in the references

加速器だから見える世界。
KEK

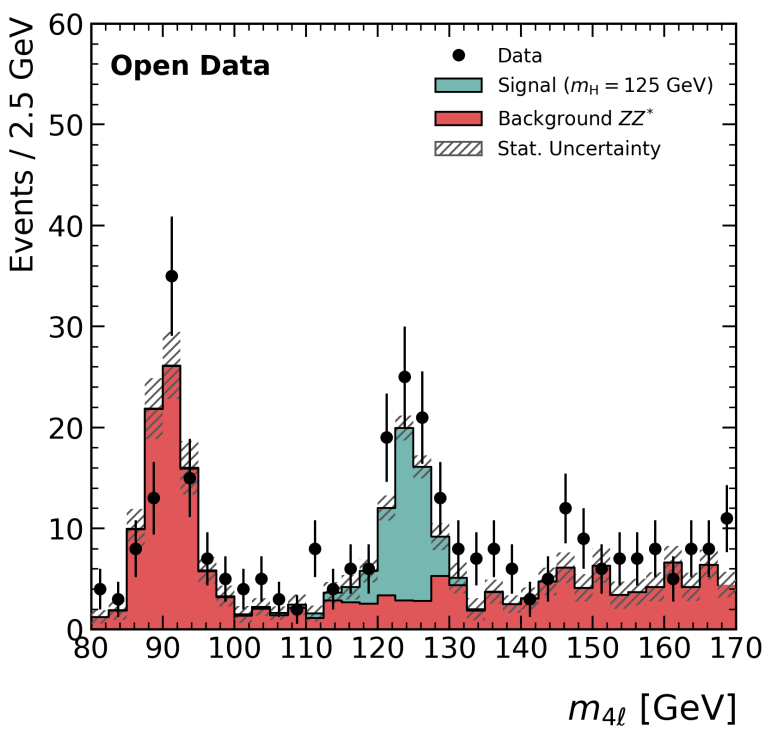
Human-reproduced results

Before automation, we manually reproduced the analysis to establish a baseline

Paper



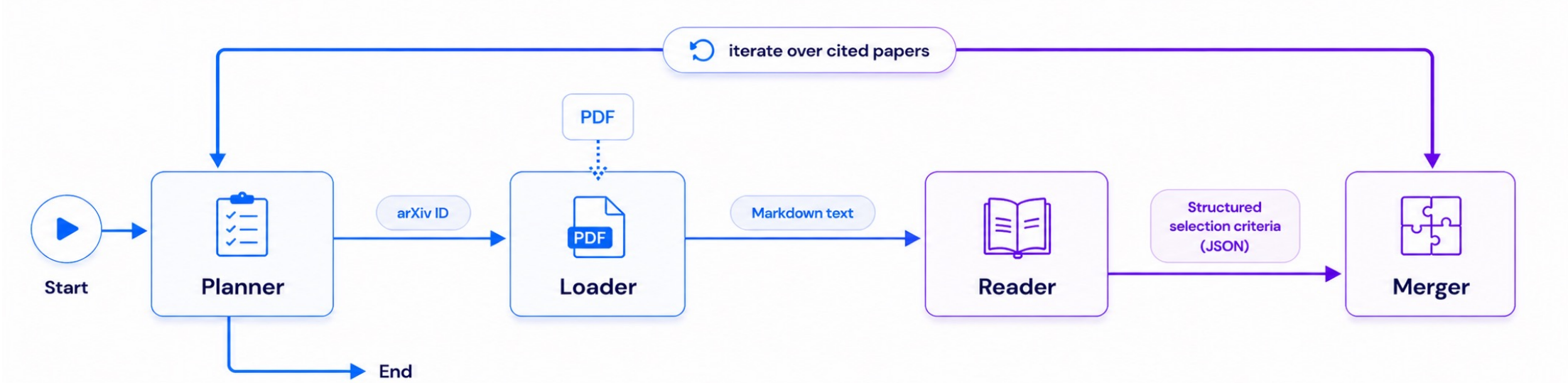
Human-reproduced



The distributions agree reasonably well — this baseline is the reference for all downstream evaluations

Step1: Selection Extraction – workflow

Goal: extract event-selection cuts from the main paper and its cited references



1. Planner

- Selects the next paper (and reference) to read

2. Loader

- Converts PDF → Markdown
- Extracts the reference list

3. Reader

- Extracts selection criteria from the text

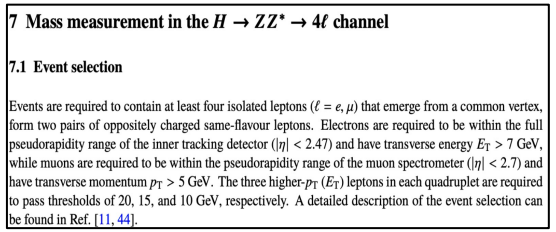
4. Merger

- Merges new criteria
- Iterates over cited papers

→ Details next slide

Step1: Selection Extraction – paper to selection criteria

Paper text (PDF)

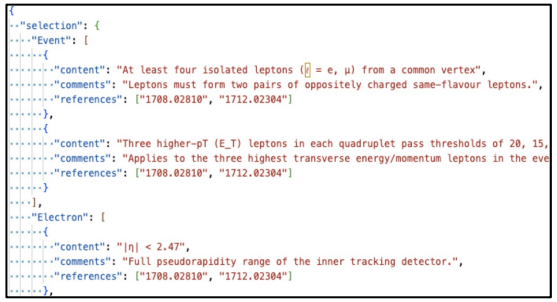


7 Mass measurement in the $H \to ZZ^* \to 4\ell$ channel

7.1 Event selection

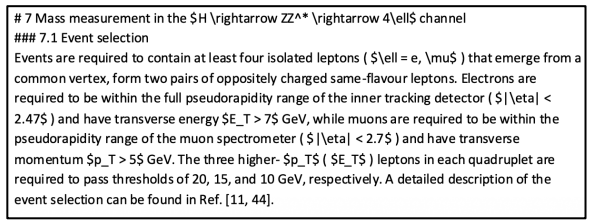
Events are required to contain at least four isolated leptons ($\ell = e, \mu$) that emerge from a common vertex, form two pairs of oppositely charged same-flavour leptons. Electrons are required to be within the full pseudorapidity range of the inner tracking detector ($|\eta| < 2.47$) and have transverse energy $E_T > 7$ GeV, while muons are required to be within the pseudorapidity range of the muon spectrometer ($|\eta| < 2.7$) and have transverse momentum $p_T > 5$ GeV. The three higher-p_T (E_T) leptons in each quadruplet are required to pass thresholds of 20, 15, and 10 GeV, respectively. A detailed description of the event selection can be found in Ref. [11, 44].

Structured LLM output




```
{
  "selection": {
    "Event": [
      {
        "content": "At least four isolated leptons (ℓ = e, μ) from a common vertex",
        "comments": "Leptons must form two pairs of oppositely charged same-flavour leptons.",
        "references": ["1708.02810", "1712.02304"]
      },
      {
        "content": "Three higher-pT (E_T) leptons in each quadruplet pass thresholds of 20, 15,
        "comments": "Applies to the three highest transverse energy/momentum leptons in the eve
        "references": ["1708.02810", "1712.02304"]
      }
    ],
    "Electron": [
      {
        "content": "|η| < 2.47",
        "comments": "Full pseudorapidity range of the inner tracking detector.",
        "references": ["1708.02810", "1712.02304"]
      },
```


Converted Markdown



7 Mass measurement in the $H \rightarrow ZZ^* \rightarrow 4\ell$ channel
7.1 Event selection
Events are required to contain at least four isolated leptons ($\ell = e, \mu$) that emerge from a common vertex, form two pairs of oppositely charged same-flavour leptons. Electrons are required to be within the full pseudorapidity range of the inner tracking detector ($|\eta| < 2.47$) and have transverse energy $E_T > 7$ GeV, while muons are required to be within the pseudorapidity range of the muon spectrometer ($|\eta| < 2.7$) and have transverse momentum $p_T > 5$ GeV. The three higher- p_T (E_T) leptons in each quadruplet are required to pass thresholds of 20, 15, and 10 GeV, respectively. A detailed description of the event selection can be found in Ref. [11, 44].

Evaluation protocol

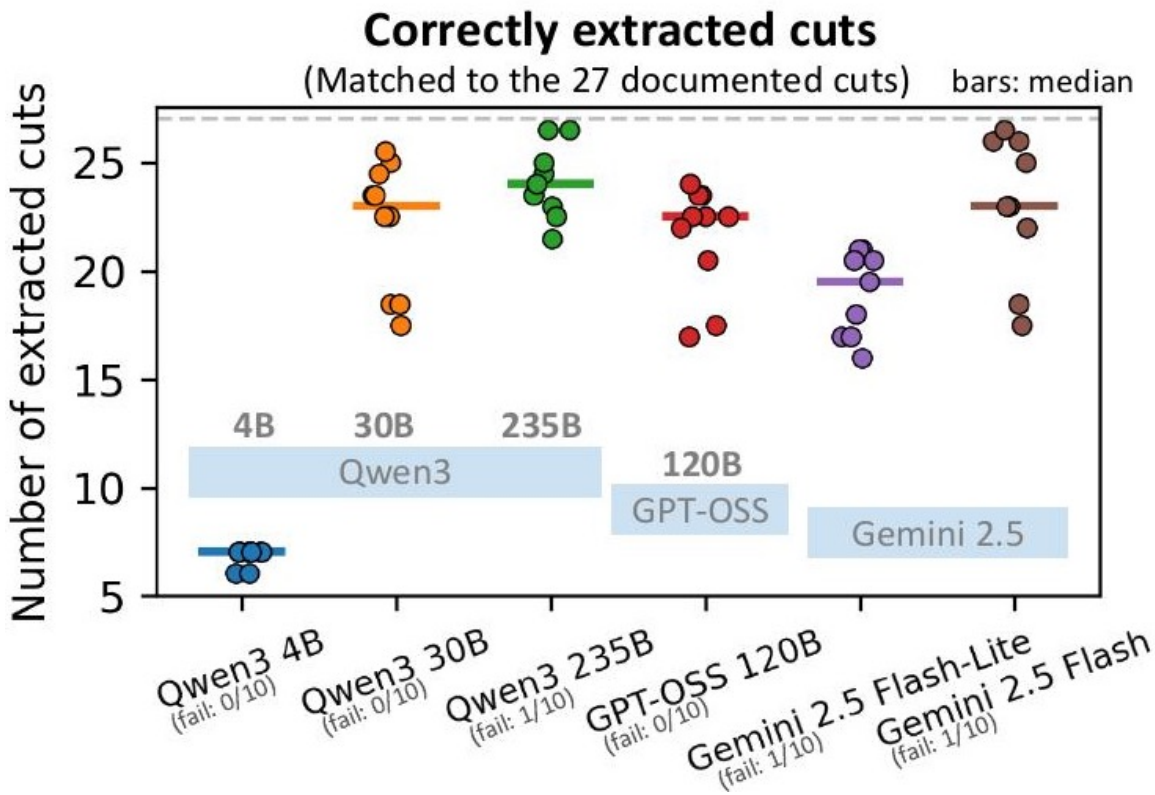
- 10 runs / model
- Compare extracted cut lists against the ground truth (27 selection cuts)
  - Score with LLM-based judges
  - Use median over repeated judge runs
  - Count contradictions to the ground truth as hallucinations

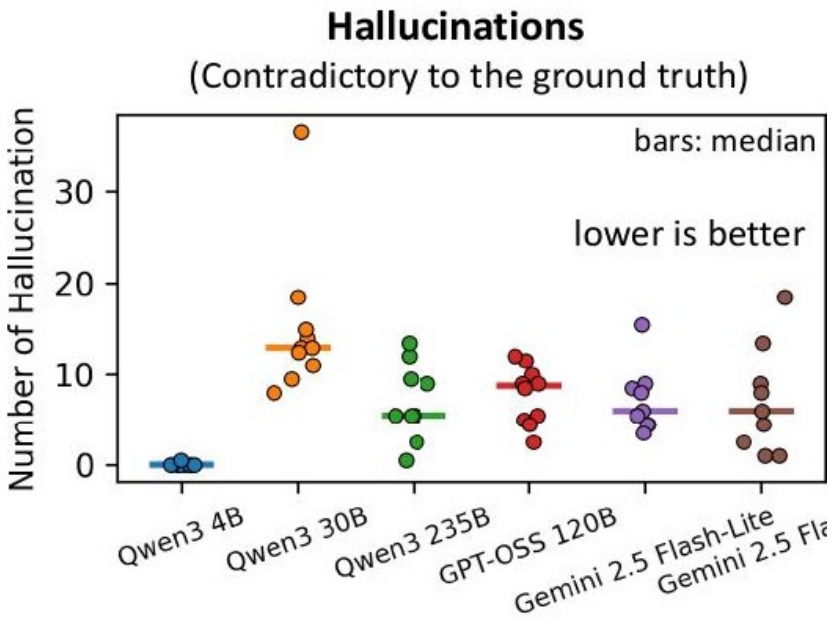
Metrics: correct cuts + hallucinations

- **Experimental setup**
  - Framework: LangChain and LangGraph, Inference backend: vLLM
  - Open-weight models: Qwen3(4B,30B,235B), Commercial models: Gemini 2.5 Flash-Lite, Flash

Primary focus is open-weight models → Local/private deployment and lower cost

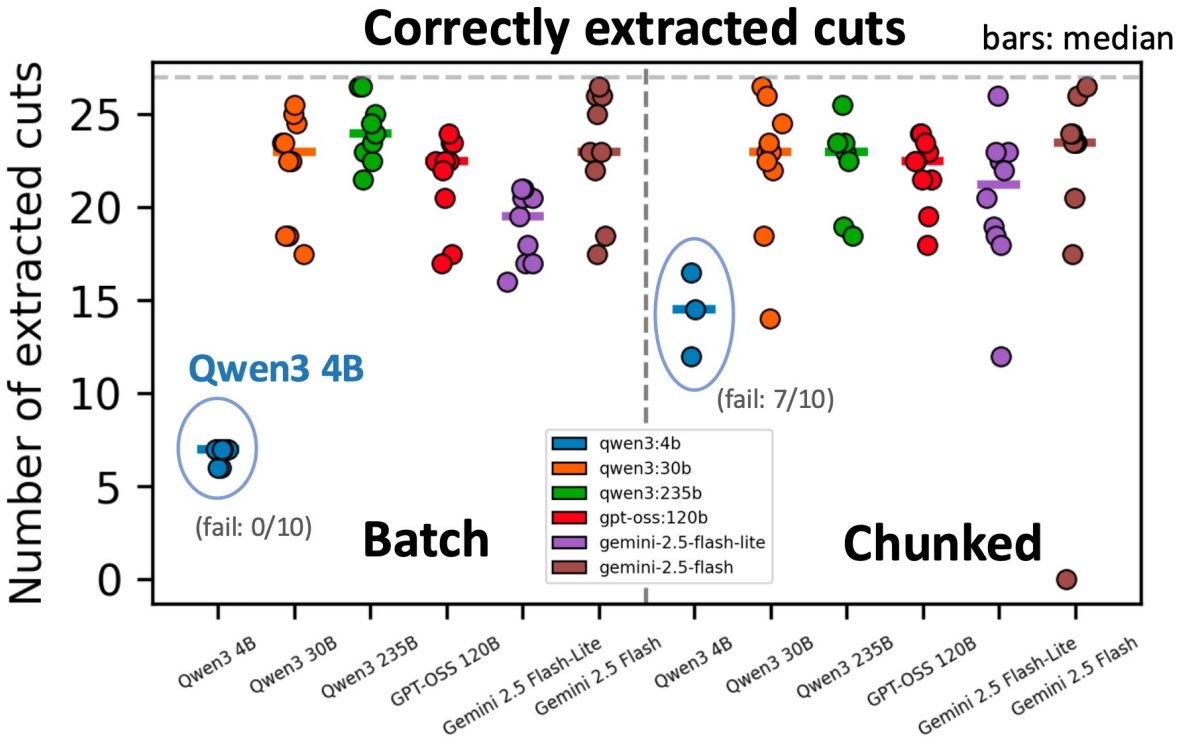
Step1: Selection Extraction – results



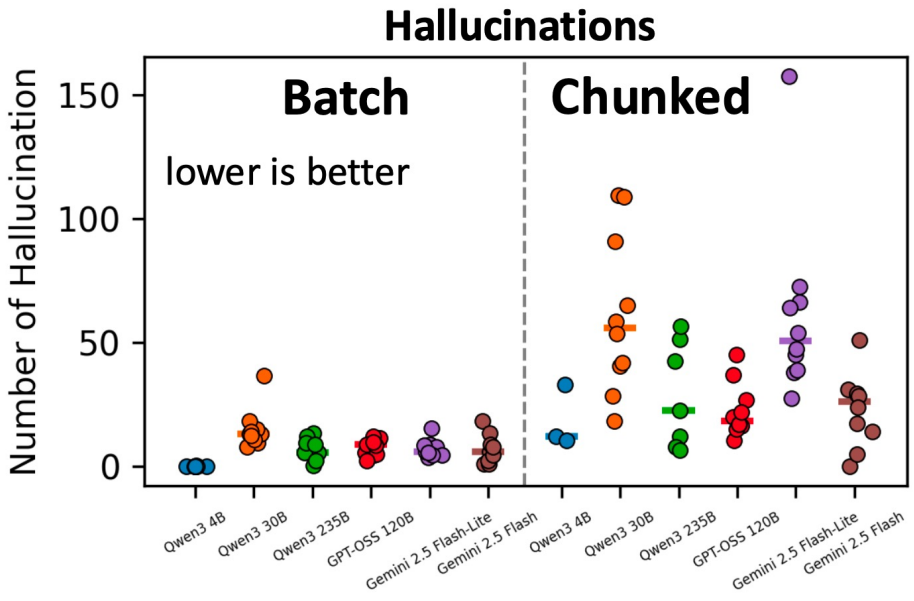


- 30B+ models extract most documented criteria; the 4B model is insufficient
- Outputs remain stochastic — larger models reduce hallucinations but are not yet fully stable

Step1: Selection Extraction – effect of chunking



Chunking: instead of processing the full paper at once, the text is split into smaller chunks and processed separately



- Chunking improves extraction for small models, especially 4B. However, it also increases hallucinations
- This trade-off remains a key challenge for practical use of small models

Step2: Code Generation – Using Coding Agents

Goal: generate executable analysis code from the extracted cuts

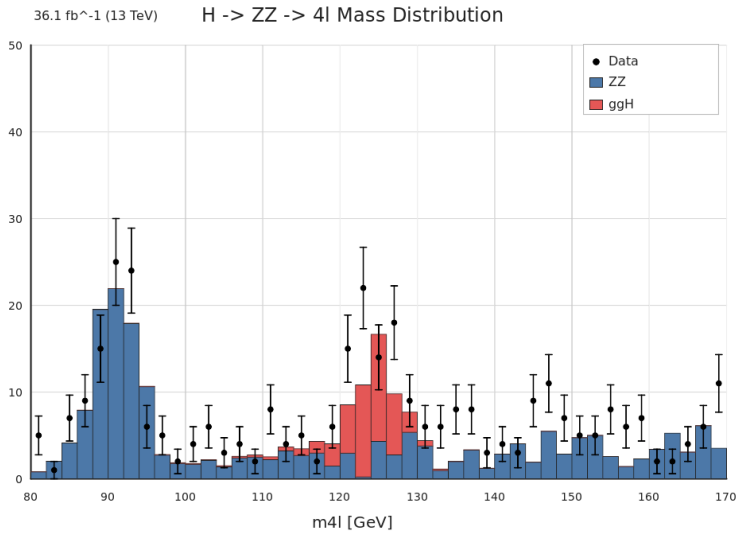
Approach

- Use off-the-shelf coding agents
  - **Claude Code and Codex**
- Provide a very simple prompt (shown on the right)
- External inputs given to the agent are only:
  - Location of the ROOT data directory
  - Event-selection list (**ground-truth cutflow so far**)
  - MC cross sections (xsec.json)
- Everything else (ROOT file structure, branch names, …) is discovered by the agent itself

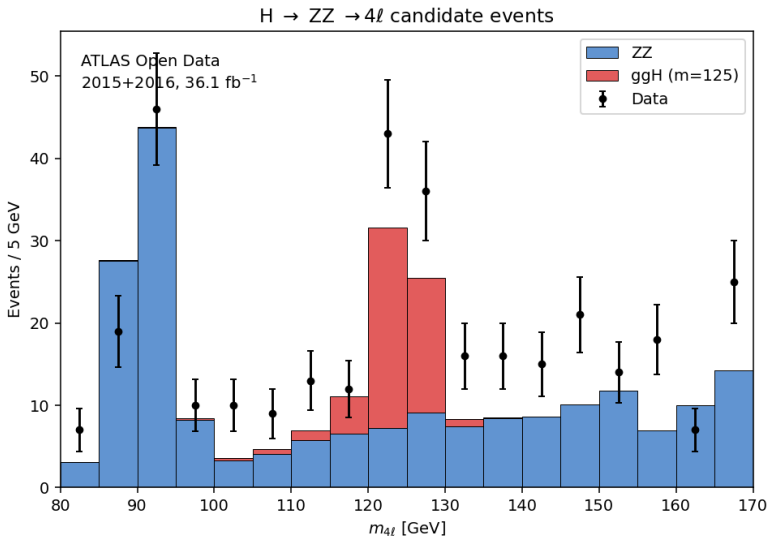
```
user@analysis:~$
Please generate code to analyze the data
in the root directory and produce the Higgs
boson mass distribution for H → ZZ → 4ℓ.

• h2zz.py performs the event classification
  described in cutflow.md and saves the
  resulting masses to masses.json.
  For MC, also save the event weights.

• plot.py reads masses.json and creates
  the mass distribution plot mass.png.
  MC scaling is read from xsec.json.

user@analysis:~$
```

Step2: Output Plots from Different Coding Agent

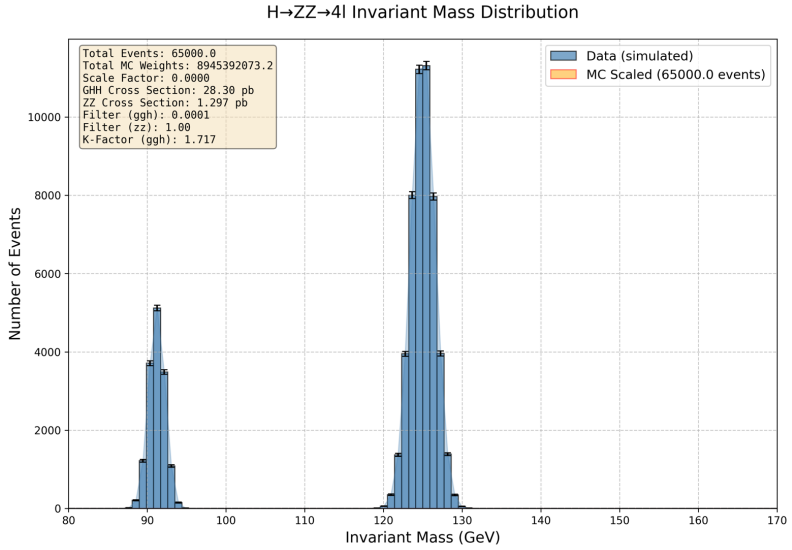
GPT5.4 (Codex)



Opus 4.7 (Claude Code)



Qwen3.5 (Claude code)



- All three agents produced runnable code and a Higgs-like mass distribution
- **The commercial models (GPT 5.4 and Opus 4.7) seem to handle the MC data correctly**
  - **At least, they appear to be able to claim a Higgs discovery**

Generating this single plot exhausted my 4-hour Claude Code token limit…

Summary and discussions

- **Goal: reproducibility framework that goes paper → code → result**
  - Benchmark: ATLAS H → ZZ* → 4ℓ, Open Data, 27 ground-truth cuts
- **Step 1 — selection extraction**
  - Custom LangChain/LangGraph pipeline extracts structured cut lists
  - Hallucinations remain an issue even for commercial models (Gemini 2.5)
- **Step 2 — code generation**
  - Delegated to coding agents (Claude Code, Codex)
  - All three agents tested produced Higgs-like distributions
- **Next**
  - End-to-end integration of Step 1 and Step 2; explore AI agents such as OpenClaw
  - Broader benchmarks beyond ATLAS H → ZZ* → 4ℓ…
  - Close the gap between open-weight and commercial LLMs, possibly with RAG