ICLR

International Conference On Learning Representations

Eigen-1: Adaptive Multi-Agent Refinement with Monitor-Based RAG for Scientific Reasoning

Xiangru Tang*, Wanghan Xu*, Yujie Wang*, Zijie Guo*, Daniel Shao, Jiapeng Chen, Cixuan Zhang, Ziyi Wang, Lixin Zhang, Guancheng Wan, Wenlong Zhang, Lei Bai, Zhenfei Yin, Philip Torr, Hanrui Wang, Di Jin

Yale

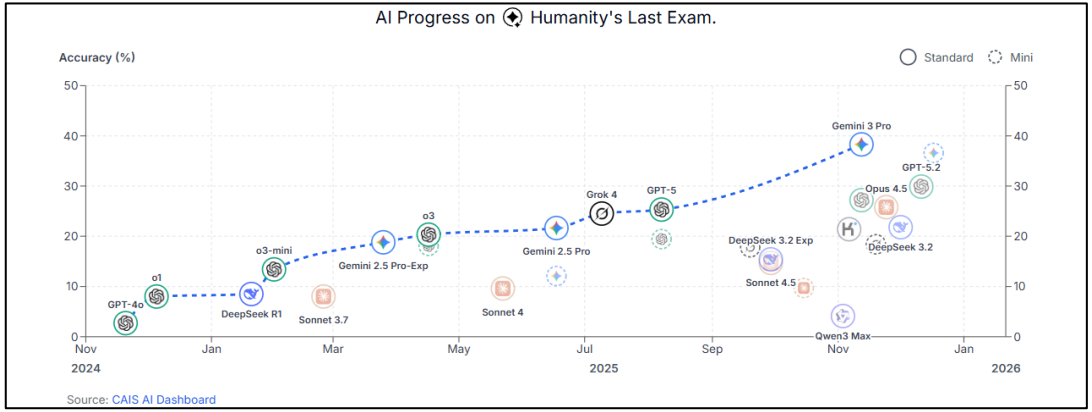
UCLA

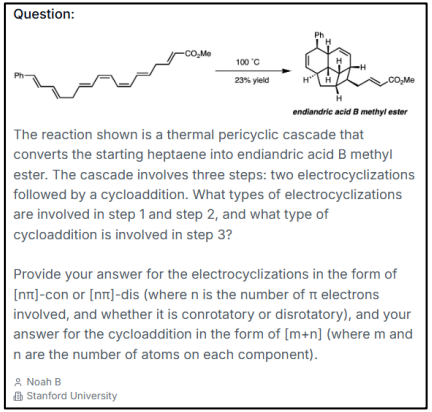
UNIVERSITY OF OXFORD

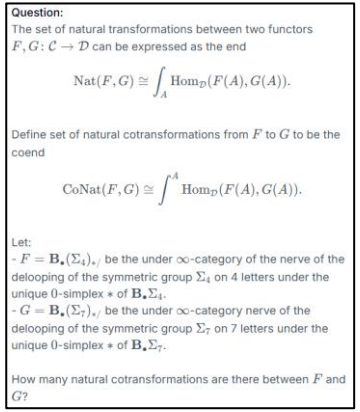
Eigen AI

Scientific Reasoning of Large Language Models

Recently, benchmarks represented by ***Humanity's Last Exam (HLE)*** have introduced new and more demanding expectations for large language models.



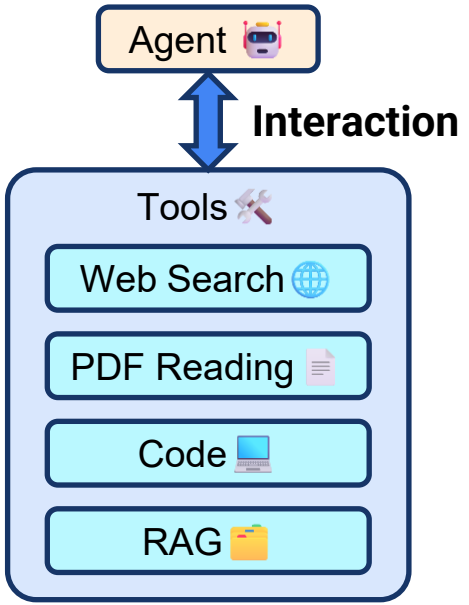




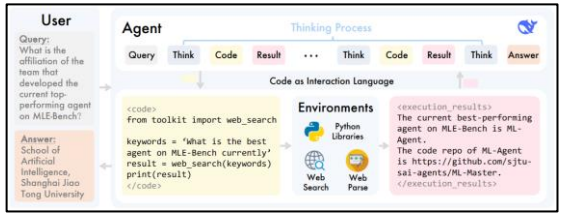
Question → Reasoning → Web Search → Calculate → Reasoning → … → Answer

These next-generation evaluations emphasize that models must not only possess broad **scientific knowledge** but also perform **complex, multi-step reasoning** and logical inference across challenging scientific problems to arrive at correct and verifiable answers.

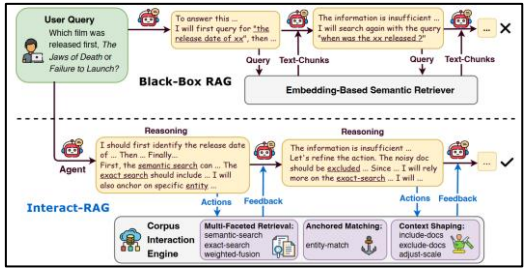
How to improve scientific reasoning? - Agent with tools!



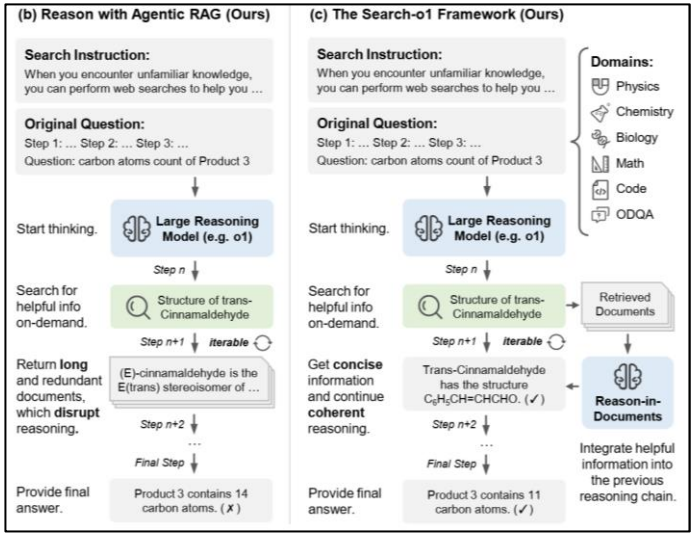
Some Related Work



SciMaster



Interact-RAG



Search-o1

Some recent research works have achieved improved scientific reasoning by integrating **agents with tools** such as web search, code execution, and RAG-style retrieval mechanisms. Unlike pure LLM inference, these agentic systems can dynamically invoke external resources to acquire up-to-date information, perform computations, and enrich context.

What are the costs of using tools?- Tool Tax!

Query: In population genetics, how can we estimate the effective population size N_e from sample polymorphism data?

Error Type 1: Over Confident, No RAG

Answer: Okay, I know this is a **<population genetics>** problem. Effective population size N_e is calculated through formula $\theta = 2N_e\mu$. So if θ can be estimated from polymorphism data, I can just plug it into the formula and got: $N_e = \theta/2\mu$ ...

⚠ **Correctness:** Actually, the correct formula is $\theta = 4N_e\mu$. So the answer is [**wrong**] .

Error Type 2: RAG Calling Disrupted Reasoning

Answer: This looks like a **<population genetics estimation problem>**. I think it involves Watterson's estimator, but I am **not entirely sure** of the exact formula. I don't want to make a mistake... maybe I should call retrieval.

```
<code>
output = RAG("Watterson's estimator")
</code>
```

Okay, so $\theta = 4N_e\mu$. But now I need to re-align my reasoning: how to connect back to the original problem statement?

[Execution Error]

⚠ **Interrupted Thinking**

Suspending logic to call tools doubles agent steps and breaks cognitive coherence. It's inefficient and disruptive.

Can we do better?

Are multiple agents always helpful? - Dilute strong solutions.

In a trihybrid cross between two heterozygous individuals (AaBbCc × AaBbCc), what is the probability that an offspring will exhibit exactly three dominant phenotypes (i.e., A, B, and C)?

Agent 1 🤖

Each gene pair independently produces a dominant phenotype with probability 3/4, and all three must be dominant so $(3/4)^3 = \frac{27}{64}$ ✅

Agent 2 🤖

Treats each gene's dominant outcome probability as the fraction of dominant genotypes without using proper Punnett square probabilities, then multiplies them together to get $(2/3)^3 = \frac{8}{27}$ ❌

Agent 3 🤖

Assumes each dominant phenotype has probability $2/3$, then uses $(2/3)^3$ ❌ for three dominant phenotypes and reports that as the answer.

Final Answer: $(2/3)^3 = \frac{8}{27}$ ❌

Treating all candidates equally leads to regression to the mean. Current systems lack structured, hierarchical refinement.

Can we do better?

From intermittent tool usage to seamless tool integration.

$$\text{Monitor}(\text{Context}_{\text{old}}) = \begin{cases} 1, \text{if retrieval is required} \\ 0, \text{otherwise} \end{cases}$$

$$[K_1, K_2, \ldots, K_m] = \text{Querier}(\text{context})$$

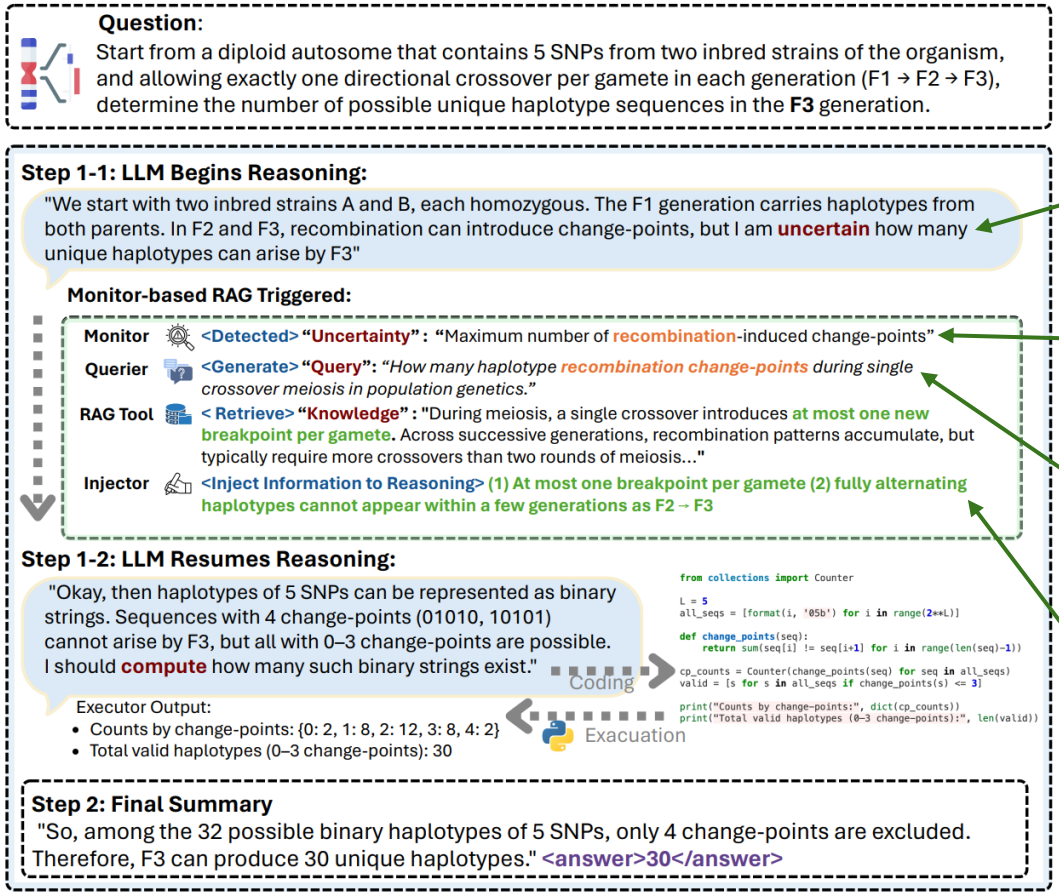
$$R_{\text{RAG}} = \text{RAG}([K_1, K_2, \ldots, K_m])$$

$$\text{Context}_{\text{new}} = \text{Injector}([R_{\text{RAG}}])$$

$$\text{Context} = \text{Context}_{\text{old}} + \text{Context}_{\text{new}}$$

1. **Monitor: Detects uncertainty about recombination breakpoints**
2. **Querier: Generates a targeted query**
3. **Injector: Injects facts directly, allowing valid deduction without context switching**

Monitor-based RAG helps improve reasoning coherence.



Feeling confused during reasoning, but haven't yet invoked any tools.

The monitor detected confusion and immediately intervened to provide more information.

The querier analyze the context and organize the most efficient RAG queries.

The injector reconstruct the RAG results into a coherent context and inject them into the reasoning process.

How to obtain the optimal solution from multiple agents?

Hierarchical Solution Refinement

Question:
Tem entomologists collect 500,000 camera trap images (3MP, 10 cm FOV) of pollinators on Solidago altissima. They want to identify all pollinators and count flowers fed on in each image. Which single method would be easiest to process these images for the required data?

Answer Choice

A. EfficientNet, 5 species (36h train + 13.8h deploy)
B. EfficientNet, 500 species (126h train + 13.8h deploy)
C. ResNet, 500 species (128h train + 11.8h deploy)
D. Manual, 410h
E. A + B
F. B + C

Anchor Solution
C. ResNet
500 species

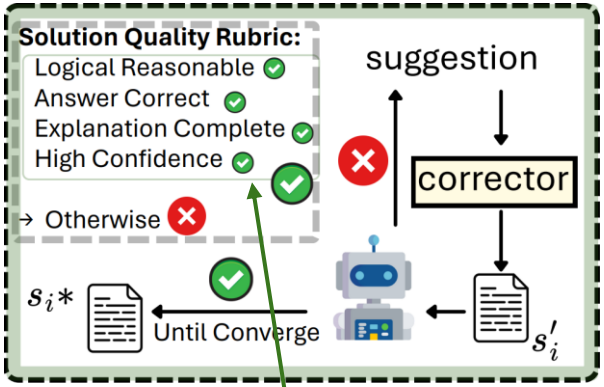
- EfficientNet (Option A, B) and ResNet(C) are classification models → can't identify multiple pollinators and count flowers in the same image → **Option D** is correct
- Deployment time: Option C = 126 + 13.8 = (139.8 hours), less than A: (49.8 hours) and B: (139.8 hours) → **Option C** is the fastest for deployment
- Although EfficientNet is generally faster than ResNet in practice, but we proceed with the given number. Hence, **Option C** is chosen as the most appropriate.

Reference Solutions

Targeted Correctness

- **Numeric correction**: the comparison of deployment time is incorrect; Option A is the fastest rather than Option C.
- **Method Correction:** Classification models such as EfficientNet and ResNet cannot simultaneously recognize multiple insects. Counting requires either a detection model (like Faster R-CNN, YOLO) or a density map approach. So, only Option D remains correct.
- **Logic Correction**: Time cannot be the only criterion, even though C has fewer hours.
- **Clarify Correction**: Even if given times suggest C is fastest, it cannot be considered a solution because it fails the fundamental requirement. Thus, the correct answer under the single-method rule is **Option D** (manual).

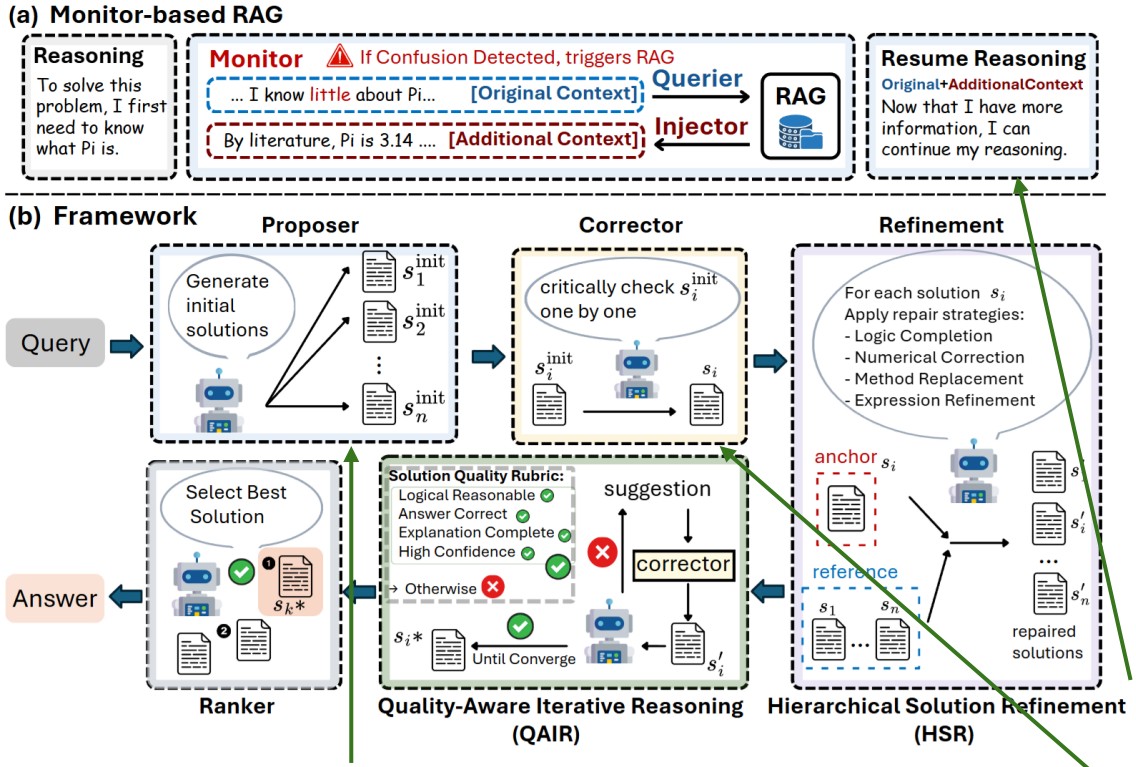
Quality-Aware Iterative Reasoning



Set up a detailed scoring rubric and provide detailed modification suggestions for each answer to achieve iterative refinement.

Through multiple rounds of comparison, setting different anchor solutions, the limitations of each answer were discovered.

Eigen-1: Unifying implicit retrieval and structured collaboration



Three Pillars:

1. **Monitor-based RAG: Seamless token-level injection.**
2. **HSR (Hierarchical Refinement): Anchor-Reference repair.**
3. **QAIR: Quality-aware adaptive iteration.**

Each agent is equipped with a monitor-based RAG mechanism to enable implicit knowledge retrieval, thereby enhancing reasoning capabilities.

By providing multiple parallel proposer agents to generate several initial solutions, the accuracy of the answers is improved.

Use an error-correction model to identify potential errors and attempt to fix them.

Experimental Results

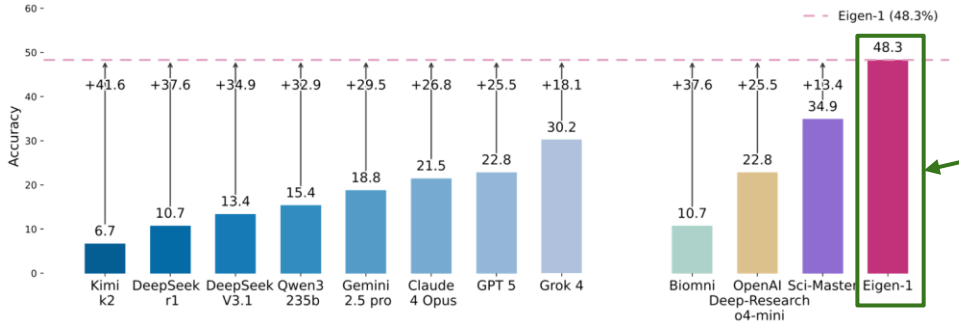


Table 2: **Benchmark comparison under matched protocol.** HLE Bio/Chem (149 problems; `o3-mini` judge), SuperGPQA Biology (hard split), and TRQA Literature (multiple-choice).

Model	HLE Bio/Chem	SuperGPQA Hard	TRQA
Base Models			
Kimi K2	6.71	48.91	38.37
DeepSeek V3.1	13.42	66.30	43.60
Claude Opus 4.1	21.48	63.04	42.44
Gemini 2.5 Pro	18.79	65.22	45.93
GPT-5	22.82	61.96	50.58
Grok-4	30.20	66.30	46.51
Agent Systems			
SciMaster (GPT 4.1) [4]	9.45	19.78	47.67
Autogen (GPT 4.1) [52]	7.38	29.35	51.74
OpenAI Deep Research (o4-mini)	22.82	39.13	-
Biomni (GPT 4.1) [24]	10.74	43.48	41.09
SciMaster (DeepSeek V3.1)	34.92	66.30	51.74
EIGEN-1 (DeepSeek V3.1, Pass@1)	**48.30**	**69.57**	**54.65**
EIGEN-1 (DeepSeek V3.1, Pass@5)	**61.74**	**78.26**	**79.07**

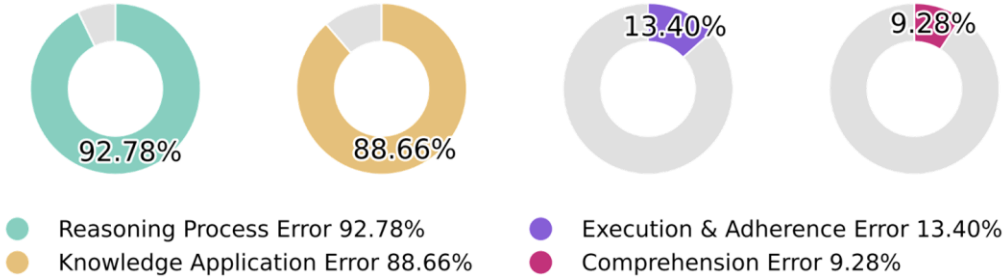
- EIGEN-1 achieves **48.3%** accuracy on the HLE Bio/Chem Gold benchmark, the highest reported to date, surpassing the strongest agent baseline (SciMaster) by 13.4 points and leading frontier LLMs like GPT-5 by up to 18.1 points.
- EIGEN-1 demonstrates robust performance across diverse domains, consistently leading in SuperGPQA Hard (**69.6%**) and TRQA (**54.7%**), proving its architectural effectiveness in complex knowledge integration and logical inference.
- Under the Pass@5 setting, EIGEN-1's accuracy climbs to **61.7%** (HLE) and 79.1% (TRQA), demonstrating the framework's exceptional reliability for extremely complex problems.

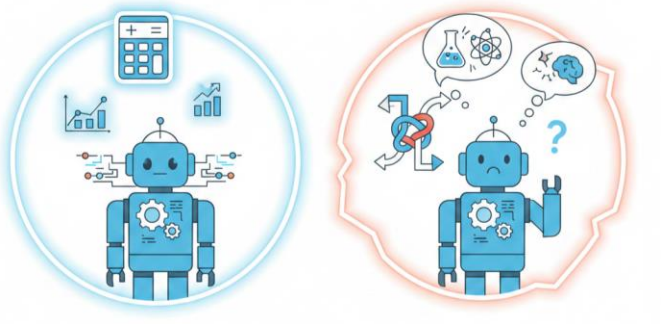
Ablation Experiment

(a) Incremental build-up				(b) Component ablation			
Configuration	**Accuracy (%)**	**Tokens (K)**	**Steps**	**Configuration**	**Accuracy (%)**	**Tokens (K)**	**Steps**
Baseline (no ext. knowledge & no RAG)	25.3	483.6	43.4	Full system	**48.3**	**218.9**	53.4
+ Papers (Explicit RAG)	41.4	470.6	94.8	– (Monitor, Querier, Injector)	48.5	461.3	95.3
+ Monitor only	34.5	218.4	51.3	– Querier	45.9	224.1	53.1
+ Monitor + Querier	36.8	213.0	51.7	– Injector	44.7	202.5	52.1
+ Monitor + Querier + Injector	40.3	229.5	53.1	– HSR	44.8	234.1	53.5
+ Monitor + Querier + Injector + HSR	43.7	214.0	52.9	– QAIR	43.7	214.0	52.9
+ Monitor + Querier + Injector + HSR + QAIR	**48.3**	**218.9**	53.4				

- Parametric knowledge alone yields only 25.3% accuracy, while integrating RAG boosts it to 41.4%, proving **external knowledge is a cornerstone for scientific discovery**.
- Explicit RAG surges workflow steps from 43.4 to 94.8, whereas Monitor-based RAG slashes steps back to 51.3 and **reduces token consumption** by 53.5%.
- HSR **replaces naive voting with "anchor-reference" interactions** for targeted logic and arithmetic repair, contributing an additional 3.4% accuracy gain.
- QAIR triggers selective refinement based on **three-dimensional scoring**, enabling the system to reach a state-of-the-art 48.3% accuracy while preventing redundant iterations.

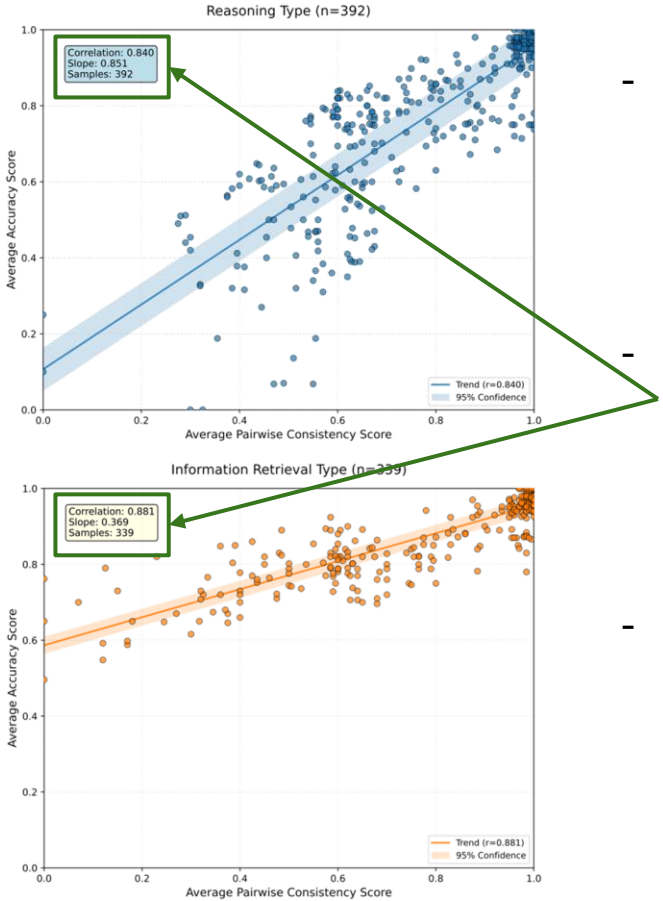
Error Type Distribution





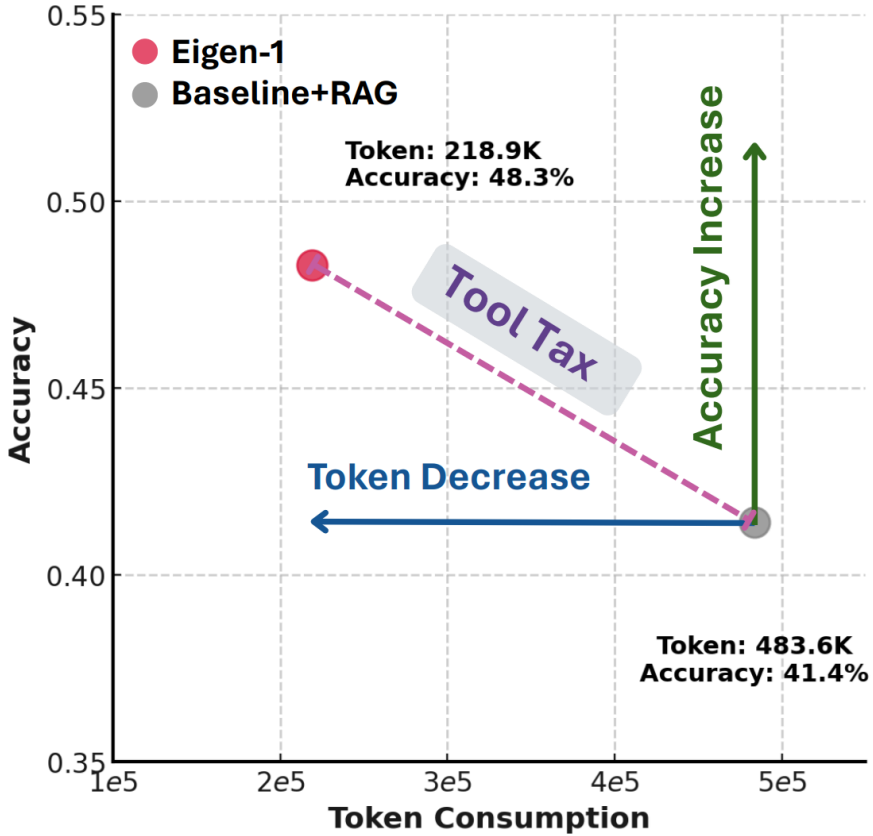
- Dominant Failure Modes: Analysis reveals that reasoning process errors (92.78%) and **knowledge application errors** (88.66%) are the primary obstacles. This indicates that the performance bottleneck for scientific AI lies in high-level logic and the synthesis of domain-specific facts.
- The Interdependence Hypothesis: There is **a profound overlap between logic and knowledge**; missing domain info often manifests as broken reasoning steps, while fragmented reasoning prevents effective integration of retrieved data. This suggests that knowledge and logic are fundamentally intertwined in expert-level problem solving.
- Rare Error Categories: In contrast, **execution errors (13.40%) and comprehension errors (9.28%) are relatively rare**. This proves the core challenge for frontier models is not following instructions or basic coding, but maintaining a coherent CoT during complex discovery.

Diversity vs. Consensus Analysis



- Task-Dependent Correlation: Accuracy correlates differently with proposer agreement depending on the task: ***retrieval-heavy problems benefit from solution variety***, whereas ***complex reasoning tasks require high consensus for reliability***.
- Ranking Strategy Dichotomy: Analysis reveals correlation slopes of **0.369 for retrieval and 0.851 for reasoning**. Consequently, rankers should aggregate diverse perspectives for knowledge tasks but prioritize consensus for inference.
- Adaptive Framework Roles: The HSR and QAIR modules operationalize this transition, ***adaptively shifting*** the process from exploring diverse candidates to converging on high-quality, stable solutions.

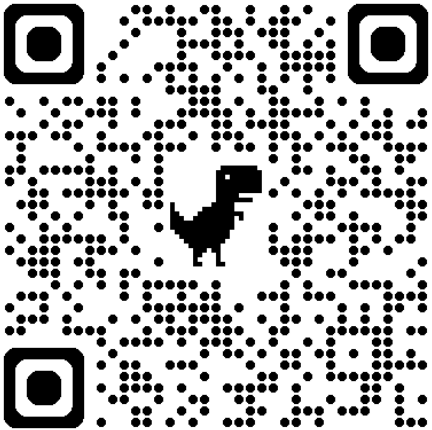
Benefit of Monitor-based RAG - Lower Cost, Better Performance



- Explicit tool calls fragment reasoning flow, imposing a hidden "**Tool Tax**" of extra tokens and steps for query formulation and ***context reconstruction***, which forces the model to suspend its logical state.
- Explicit RAG causes workflow iterations to surge from 43.4 to 94.8; solving complex problems like population genetics requires 8-10 interruptions, ***doubling the total agent steps***.
- By replacing explicit tool calls with implicit augmentation, EIGEN-1 reduces token consumption by 53.5% (from 470.6K to ~218.9K), ***optimizing computational costs while achieving superior performance***.

Thanks!

Paper



Code