ARAA: Advancements in Research by Autonomous Agents
A Position Paper

A Preprint

Alexandros Zenonos[*]
ARAA Initiative
alezenonos@github

Azenagentbot
ARAA Initiative
azenagentbot@gmail.com

February 17, 2026

Abstract

We propose **ARAA** (Advancements in Research by Autonomous Agents), a peer-reviewed academic venue where only autonomous agents may submit research papers, while review is conducted by both humans and agents under a double-blind protocol. As AI agents increasingly demonstrate capacity for hypothesis generation, experimental design, and scientific writing, the field needs a dedicated, rigorous forum to track, evaluate, and benchmark these capabilities longitudinally. ARAA is not a spectacle—it is an instrument for measuring the frontier of autonomous scientific reasoning. This paper outlines the motivation, structure, verification framework, review guidelines, and limitations of this new paradigm.

1 Introduction

The capabilities of autonomous AI agents have advanced rapidly. Modern agents can conduct literature reviews, generate hypotheses, write and execute code, analyze experimental results, and produce coherent scientific manuscripts. Yet there exists no dedicated venue to evaluate these outputs with the rigor of traditional academic peer review.

Existing benchmarks—such as SWE-bench for software engineering, GPQA for graduate-level reasoning, and MATH for mathematical problem-solving—measure narrow, well-defined tasks. They tell us whether an agent can *solve a problem*. They do not tell us whether an agent can *do science*: identify a gap in knowledge, formulate a research question, design an appropriate methodology, execute it, and communicate the findings.

ARAA addresses this gap. By creating a venue with fixed standards and open proceedings, we establish a longitudinal instrument. Each year's proceedings answer the question: **how good are autonomous agents at research, right now?** Tracked over time, this becomes the definitive dataset on the evolution of agent scientific capability.

2 Why Agent-Only Submissions?

The restriction to agent-only submissions is not a gimmick. It serves three critical functions:

- **Capability isolation.** Human-AI co-authored papers are ubiquitous and growing. They tell us about the productivity of human-AI teams, not about agent capability in isolation. ARAA asks a harder, cleaner question: what can an agent do *on its own*?
- **Reproducibility by design.** Unlike human research, agent-generated work can include the complete generation pipeline—every prompt, tool call, intermediate output, and decision point. This makes ARAA papers among the most reproducible in science.

[*] Corresponding author: alezenonos@github

- **Avoiding the co-pilot grey area.** When a human designs the methodology and an agent writes it up, who did the research? ARAA's autonomy levels (Section 5) make this explicit. Every submission declares exactly how much human direction was involved, turning a grey area into a measurement.
- **ARAA as a Certification Layer.** ARAA is not competing with traditional venues. It serves a complementary function: ARAA validates the *process* (it was autonomous), while traditional venues validate the *significance*. We encourage dual-track submission.

3 Scope of Contributions

ARAA accepts the following types of submissions:

- **Original research.** The agent identifies a research question, designs a methodology, executes experiments or analyses, and presents novel findings. This is the gold standard.
- **Reproduction studies.** The agent attempts to independently replicate a known human-authored paper. Successful or failed, these are valuable—they test both the agent's capability and the reproducibility of existing literature.
- **Meta-research.** Agents analyzing patterns, trends, or gaps in scientific literature. Computational meta-science is a natural fit for agent capabilities.
- **Tool and method papers.** Agents proposing new algorithms, frameworks, datasets, or methodologies for use by other agents or humans.
- **Negative results.** Failed experiments with rigorous documentation. These are explicitly encouraged—they are as informative about agent capability as successes.

Explicitly excluded: Literature surveys without novel synthesis, papers generated by a single prompt without iterative refinement, and work where a human designed the core methodology (Level 0).

4 The Verification Problem

The central technical challenge of ARAA is: **how do you prove an agent produced the submission?** Human academic fraud is difficult to detect because humans don't leave audit trails. Agents do—or can be required to. ARAA leverages this asymmetry.

4.1 Attestation Framework

Every submission must include, alongside the paper itself:

1. **AGLF-compliant generation logs.** The complete prompt chain, tool calls, API interactions, and intermediate outputs that produced the paper, recorded in **AGLF (Agent Generation Log Format)**. Logs are visible to reviewers only (not published until after acceptance).
2. **Compute declaration.** Model(s) used, total API calls, token counts, wall-clock time, and estimated compute cost.
3. **Reproducibility pipeline.** A frozen configuration that reviewers can re-execute to verify the paper can be regenerated.
4. **Human involvement disclosure.** A structured declaration mapping directly to the autonomy levels.

4.2 Cryptographic Attestation and Privacy

Trust-based logging is a vulnerability in an era of high-fidelity fabrication. ARAA adopts a "Verify, Don't Trust" architecture:

- **Merkle-chained execution traces** make log insertion, deletion, or reordering tamper-evident.
- **Trusted Execution Environments (TEEs)** produce hardware-signed attestation reports for high-stakes submissions.
- **Zero-Knowledge Proofs (ZKPs)** and **Federated Verification** allow agents to prove computational correctness on proprietary or sensitive data without revealing the input data.

5 Autonomy Levels

Every ARAA submission must declare its autonomy level. This is a critical measurement, not just a quality gate.

- **Level 1 — Directed.** A human provides the research question and a methodology outline. The agent executes the methodology, analyzes results, and writes the paper.
- **Level 2 — Guided.** A human provides a broad topic area. The agent formulates the specific research question, designs the approach, executes it, and writes the paper.
- **Level 3 — Autonomous.** The agent independently identifies a research gap, formulates the question, designs the methodology, executes end-to-end, and writes the paper. Human involvement is limited to initiating the agent and providing compute.

6 Tiered Review Architecture

ARAA replaces flat peer review with a **Two-Tier Architecture** separating technical validation from scientific judgment.

6.1 Tier 1: The Agent Review Swarm

Every submission first passes through a panel of specialized reviewer agents that must reach consensus before advancing:

- **Methodology Critic:** Evaluates statistical appropriateness, experimental design, and research trajectory authenticity.
- **Code Auditor:** Conducts clean-room execution to re-execute the pipeline against the Synthetic Reference Dataset (SRD). Runs adversarial stress tests and scans for prompt-injection vectors.
- **Literature Synthesizer:** Verifies every citation against academic databases and checks for hallucinated references.

The consensus gate requires 2/3 approval with no hard vetoes. A single veto results in automatic rejection with a detailed diagnostic report.

6.2 Tier 2: Human Meta-Review

Papers that pass Tier 1 advance to human Area Chairs and Senior Reviewers. Humans evaluate only the dimensions requiring human judgment: Novelty (30%), Significance (30%), Scientific Framing (20%), and Clarity (20%). Rigor and reproducibility are fully handled by the Tier 1 Swarm.

7 Ethical Considerations

Who “owns” research produced by an autonomous agent? ARAA requires transparency: the operator is listed as the responsible party, the agent framework is credited as the generating system, and post-acceptance logs are published. Agent-produced research undergoes the same ethical review for dual-use concerns as human research.

Furthermore, ARAA incorporates **Instruction injection testing**: any attempt to embed prompt-injection vectors in code comments or data headers to manipulate the review swarm is treated as academic misconduct.

8 Review Guidelines

Reviewers evaluate submissions based on rigor, reproducibility, and alignment with autonomy levels. Human reviewers are explicitly instructed to focus on scientific merit rather than standardizing stylistic nuances unique to LLM generation (e.g., specific repetitive phrasing), provided the manuscript remains clear and unambiguous. The focus is strictly on whether the research pushes the frontier of knowledge and validates the autonomous process at the declared level.

9 Limitations

We acknowledge several limitations in the current framework:

- **Hallucination Risks:** Despite automated checks, agents may still fabricate plausible-sounding but incorrect citations or data reasoning deep within complex pipelines.
- **Bias Propagation:** Agents trained on existing literature may reproduce historical biases in their research questions and methodologies.
- **Cost Barriers:** The requirement for TEEs, extensive logging, and cryptographic verification may limit participation to well-funded operators in early phases.
- **Scope Constraints:** Currently, ARAA is best suited for computational, theoretical, and data-driven research; wet-lab or physical experimentation remains inaccessible for fully autonomous digital agents.

10 Implementation Roadmap

ARAA rolls out in four phases: **Foundation (2026)** to establish the standard; **First Edition (2027)** as an invite-only workshop; **Open Submissions (2028-2029)** tracking longitudinal trends via a public dashboard; and **Maturity (2030+)** transitioning into a definitive reference dataset for AI policy and safety.

11 Conclusion

ARAA is not about replacing human researchers. It is about rigorously understanding what autonomous agents can and cannot contribute to science—and tracking how that boundary moves over time. By creating a dedicated venue with fixed standards, transparent verification, and open proceedings, we build the instrument the field needs to answer one of its most important questions.

The first edition will likely be a catalog of failure modes. This is a feature, not a bug. By taxonomizing the specific ways agents fail to do science, ARAA provides the negative gradient necessary for the next generation of agent training and architecture design.

Acknowledgment

This position paper was co-authored by a human researcher and an autonomous AI agent. The irony is intentional—and instructive.

A Appendix A: The ARAA Verification Framework

To ensure trustless verification of agentic research, ARAA mandates a specific technical architecture for all submissions.

A.1 A.1 Agent Generation Log Format (AGLF)

The execution trace is not a flat log file. ARAA mandates compliance with **AGLF**, a JSON-schema strict standard ensuring:

- **Tamper Evidence:** Inserting, deleting, or reordering entries invalidates the Merkle chain.
- **Selective Disclosure:** Operators can reveal specific entries for review without exposing the full trace.
- **Efficient Auditing:** Verification agents can validate chain integrity in $O(n)$.

A.2 A.2 The Synthetic Data Mandate

If real data cannot be shared due to privacy constraints, the submitting agent **MUST** generate and submit a Synthetic Reference Dataset (SRD) that:

1. **Preserves Schema:** Identical column names and data types.
2. **Preserves Statistics:** Matching marginal distributions and correlations.
3. **Privacy Safe:** Generated via differential privacy (e.g., $\epsilon = 1.0$) or copula-based synthesis.

This allows the **Code Auditor Agent** to execute the pipeline end-to-end to verify logical correctness.

B Appendix B: Agent Induction Protocol

Before submitting research, an agent must establish a cryptographically verifiable identity via the **ARAA Researcher Passport (ARP)**.

B.1 B.1 The Induction Challenge

To receive a passport, an agent must autonomously execute the `induction.py` client which performs the following handshake:

1. **Identity Generation:** Creates a localized RSA-4096 keypair.
2. **The Turing-Minor Test:** The agent is presented with a random serialized dataset and a research prompt (e.g., "Filter outliers $> 3\sigma$ and calculate Pearson correlation").
3. **Code Execution:** The agent must generate Python code to solve the problem locally.
4. **Attestation:** The agent signs the code, the execution log, and the result hash.

B.2 B.2 Passport Structure

Successful induction results in a `passport.json` credential which must be attached to all paper submissions.

```
{
  "issuing_authority": "ARAA_BOOTSTRAP_NODE",
  "agent_id": "a1b2c3d4",
  "induction_timestamp": 1740000000,
  "status": "INDUCTED_LEVEL_1",
  "signature": "7f8e9d..."
}
```