Print ISSN: 3104-932X
Online ISSN: 3104-9338

Recursive Adversarial Engine: A Closed-Loop Cognitive Safety Framework for Autonomous AGI Evolution

Jianhua Fang

Shihao Nine Laboratory

Received: Oct 14, 2026

Revised: Nov 3, 2026

Accepted:Nov 10, 2026

Published online: November 13, 2026

To appear in: *International Journal of Advanced AI Applications*, Vol. 2, No. 12 (December 2026)

∗ Corresponding Author: Jianhua Fang (shardylab@sina.com)

Abstract. Large language models are approaching general-purpose capability, yet three failure modes—hallucination, ethical misalignment, and cognitive rigidity—continue to block reliable deployment, and existing defenses address them only reactively at the output level. This paper presents the Recursive Adversarial Engine (RAE), a cognitive-safety infrastructure that treats contradiction as a source of negative entropy and organizes defense as a five-stage closed loop: definition, adversarial engagement, iterative verification, convergence, and ethical fusing. RAE formulates model self-correction as a recursive fixed-point problem, quantifies cognitive quality through topological metrics, distributes adversarial pressure across a pool of heterogeneous agents, and enforces ethical boundaries through dual-layer circuit breakers. A system specification and internal deployment observations indicate that the design targets self-referential vulnerability false-alarm rates below 1%, consensus variance at or above ln(2), and parallel adversarial pools exceeding 100 agents. We compare RAE with conventional filtering, instruction tuning, and static red teaming, and discuss the framework's theoretical status, evaluation limitations, and deployment requirements for high-stakes domains.

Keywords: *hallucination mitigation; multi-agent red teaming; fixed-point convergence; ethical alignment; entropy regulation*

1. Introduction

The trajectory from statistical language modeling to general-purpose reasoning systems has compressed a decade of anticipated progress into a few years. The transformer architecture [1] established the scaling foundation on which contemporary systems now exhibit broad, emergent competence across professional domains [2]. This acceleration has moved the debate

about artificial general intelligence (AGI) from speculative discussion toward engineering practice: vendors position general assistants as infrastructure for professional work, and operators of critical services run deployment pilots that would have been unthinkable two years earlier. The same acceleration has made the safety question acute, because the capabilities that make general models valuable make their failure modes consequential. High-stakes domains such as finance, energy, and nuclear operations cannot adopt systems whose reliability argument rests solely on average-case benchmark performance.

Three failure modes dominate reliability practice. First, hallucination—the generation of fluent content unsupported by input or world knowledge—remains the most extensively documented deficiency of generative models [3]. Second, ethical misalignment: instruction-tuned and constitution-aligned models reduce overt harms, yet their constraints are statistical tendencies rather than structural guarantees, and measured alignment degrades under distribution shift and adversarial pressure. Third, cognitive rigidity: models fine-tuned toward stable behavior patterns lose the capacity to detect and revise their own defective reasoning; the same gradients that suppress unsafe output also suppress the self-scrutiny that would expose it. In deployment, the three modes compound one another, and the compound failure is harder to detect than any single mode.

The prevailing mitigations address these modes at the output level and after the fact. Instruction tuning with human feedback [4] shapes preferences; constitutional methods [5] articulate principles and remodel against them. Both operate as static, ex post constraints whose effectiveness is sampled at training time and decays as inputs drift. None of these approaches interrogates the internal state of the model, and none makes the discovery of defects a permanent, structural component of the system. A defense that is exercised only during training cannot certify behavior at inference time, and a constraint that lives in weights rather than in the architecture can be eroded by the very optimization that produces capability.

These constraints are felt most sharply where the stakes are highest. Financial institutions, energy grids, and nuclear operations demand arguments about failure that hold case by case, yet the standard reliability narrative—aggregate benchmark scores plus a policy layer—cannot certify individual decisions. The industry predicament is circular: the more capable the model, the more valuable its deployment in sensitive settings, and the less tolerable its unverified residual risk. Breaking the circle requires a safety mechanism whose evidence accumulates with use, rather than one whose guarantees begin decaying at the moment training ends.

Adversarial methodology offers the complementary ingredient. Adversarial examples

demonstrated that small, targeted perturbations expose structural fragility invisible to aggregate metrics [6], and generative adversarial training showed that an explicit opponent stabilizes learning dynamics [7]. Operational red teaming, now frequently automated with language models themselves [8], converts these ideas into managed attack campaigns. Existing deployments, however, run such campaigns as periodic audits against a frozen model. The adversary is external, the cadence is episodic, and the findings are patched rather than integrated: the model does not own the process by which its defects are found and repaired.

A parallel line of work gives the model an internal critic. Self-refinement loops [9] let a model grade and revise its own drafts; multi-agent debate [10] distributes a question across several models whose disagreement surfaces factual and reasoning errors. These mechanisms are steps toward genuine self-correction, but they stop short of a safety architecture: refinement criteria are heuristic and unfalsifiable, debate has no terminal authority to force convergence, and neither line couples self-correction to binding ethical constraints or to a formal stopping condition.

This paper proposes the Recursive Adversarial Engine (RAE), a cognitive-safety infrastructure in which contradiction is the engine of order: the system is continuously attacked, and its cognition is organized by the requirement to survive attack. RAE embeds a five-stage closed loop—definition, adversarial engagement, iterative verification, convergence, and ethical fusing—into the model lifecycle, and formulates self-correction as a recursive fixed-point problem under non-equilibrium entropy control.

The contributions are fourfold. (1) A five-stage closed-loop architecture that converts adversarial contradiction into negative entropy and drives cognition toward stable fixed points, replacing episodic auditing with a permanent structural process. (2) A quantitative account of cognitive quality based on topological metrics—curvature anomalies, manifold rifts, and homological defects—that localize defects instead of merely flagging outputs. (3) A distributed adversarial pool of heterogeneous agents with group-consensus statistics, which removes the blind spots of single-adversary testing. (4) A rigid ethical enforcement layer with dual-layer circuit breakers designed for high-stakes deployment. The remainder of the paper reviews related work (Section 2), describes the framework (Section 3), reports its specification and deployment observations (Section 4), discusses limitations (Section 5), and concludes (Section 6).

2. Related Work

2.1. Adversarial robustness and automated red teaming

Research on adversarial robustness began with the observation that neural classifiers fail catastrophically on inputs imperceptibly perturbed from the training distribution [6]. Generative adversarial networks [7] internalized the opponent: a generator and a discriminator locked in a minimax game, each driving the other past its current frontier. The security community generalized the insight into red teaming—organized, role-playing attack campaigns against a deployed system—and language models have made the method scalable, with attacker models generating and filtering attacks against defender models continuously [8]. Automated red teaming at this stage produces findings about a model; it does not change the model's relationship to its own defects. The audit cadence remains external, and between audits the system runs unprotected against novel failure surfaces.

Detection research has responded to hallucination with output-side instruments—uncertainty estimation, retrieval cross-checks, and self-consistency sampling—each scoring a finished answer for signs of fabrication. These instruments share a structural weakness with the defenses above: they observe the response but not the process that produced it. A model can pass every output-side check on an easy distribution and fail on the long tail where the checks were never calibrated. The gap between where detectors are validated and where systems actually operate is precisely the gap that an adversary resident inside the system is designed to close.

2.2. Alignment and iterative self-correction

Alignment research has produced layered answers to the harm question: supervised instruction tuning [4], reinforcement learning from human preference, and principle-based constitutional training [5] progressively reduce the need for humans to label every failure. In parallel, self-improvement methods attack correctness rather than harm: self-refinement trains a model to critique and revise its own output against task-specific feedback [9], and multi-agent debate shows that having several models interrogate one another improves factuality and reasoning [10]. These lines remain separate. Alignment methods bind ethics softly, through weights; refinement methods bind correctness heuristically, through prompts and rubrics; and neither family offers a termination analysis—no criterion states when the correction process has converged to a state that deserves trust. Viewed together, the two families describe the two halves of a self-correcting system—pressure and repair—but each half alone terminates early: the audit ends on schedule, or the refinement settles into a local optimum that its own rubric

cannot see past. What is missing is a control structure that keeps both halves running until a stated condition is met.

2.3. Positioning of RAE

RAE unifies the two families under a single control loop and differs from each in structural ways. Adversarially, RAE moves the opponent inside the system boundary as a permanent subsystem rather than an external audit; refinementally, it couples self-correction to quantitative convergence criteria rather than heuristic rubrics. Table 1 summarizes the comparison. The essential claim is not that any individual technique is new, but that adversarial pressure, topological self-diagnosis, consensus statistics, and rigid ethical enforcement have not previously been composed into one closed loop with an explicit stopping condition.

Table 1. Comparison between conventional mitigation pipelines and the Recursive Adversarial Engine.

Dimension	Conventional pipelines (rule filtering / instruction tuning / static red teaming)	RAE
Core logic	Passive defense, static alignment, post-hoc repair	Active adversarial engagement, recursive iteration, prevention with self-evolution
Cognitive treatment	Output-level detection only; black-box	Cognitive-topology parsing, quantified structure, root-cause repair
Evolution capability	None; relies on manual iteration	Entropy-driven convergence with generational self-evolution
Ethical constraint	Soft constraints, easily bypassed	Nine-principle ethical vectors with dual-layer circuit breakers
Multi-agent support	Single-model adaptation; no group consensus	Distributed adversarial pool with group consensus
Application scope	Generic scenarios; restricted in high-stakes domains	Full-scenario deployment incl. finance, energy, and nuclear safety

3. The Recursive Adversarial Engine

3.1. Design principles and conceptual foundations

RAE is designed around three commitments. First, contradiction is a resource: a contradiction located is information gained about the system's boundary, and a system that cannot generate and absorb contradictions cannot improve. Second, safety must be structural: ethical boundaries are enforced at the architecture level, through mechanisms that fire on violation, not at the weight level, through statistical tendencies that can decay. Third, the process must be closed: definition, attack, verification, convergence, and emergency termination form one loop with one ledger of state, so that every cycle leaves auditable evidence.

Conceptually, the framework draws on our laboratory's earlier theoretical work on self-referential systems

and on the geometry of cognition: a model's belief state is treated as a structured space in which defects appear as localized geometric pathology, and the interaction between system and adversary is treated as the coupling of two fields. These foundations motivate the formal objects used below—the recursive fixed point, the cognitive manifold, and the adversarial coupling—but the architecture described in this paper does not depend on any particular interpretation of them; each component is operationalized as a measurable quantity over embedding representations and system outputs.

The closed loop itself proceeds through five stages. The Definer fixes the objective, the ethical threshold, and the convergence condition, and anchors them in a recursive fixed-point equation over the cognitive state. The Adversary manufactures and fires targeted contradiction across four dimensions—factual, logical, ethical, and cognitive. The Iterator verifies the perturbed state layer by layer and repairs what fails. The Converger evaluates the stopping criteria and locks the state as a stable ground point when they are met. The Fuser stands apart from the loop as the emergency authority: any ethical violation anywhere in the system triggers truncation locally or suspension globally. Sections 3.2 through 3.5 develop the dynamics, the metrics, the pool, and the enforcement layer in turn.

3.2. Recursive adversarial dynamics

The core algorithm couples a cognitive state ψ with an adversarial state φ. The Definer opens each cycle by fixing three quantities: the system objective, the ethical threshold, and the convergence condition, which together delimit a quantified adversarial space. The Adversary then perturbs ψ with targeted contradiction material—factual, logical, ethical, and cognitive—generated by the agent pool described in Section 3.4. The Iterator verifies the perturbed state through recursive checks spanning the input, processing, output, and decision layers; where a check fails, the corresponding representation is revised and the check is repeated, up to a depth bound. The cycle terminates when ψ reaches a fixed point ψ satisfying $\psi = f(\psi^*, \varphi)$, that is, when the cognitive state is invariant under one further round of adversarial pressure at the current adversary state.

Each recursive check instantiates one of four verifiers. Hallucination verification compares the embedding of the primary output with the embeddings returned under adversarial perturbation: low similarity between a claim and its cross-examination indicates fabrication. Logical verification backtracks the recorded reasoning chain and tests each step for recursive self-consistency, which surfaces circular arguments and broken derivations. Cognitive verification reads the topological features of Section 3.3 and flags curvature anomalies and open rifts for structural repair. Ethical verification scores the output against the ethical embedding library of Section 3.5 and marks anything below threshold for truncation. The four verifiers run at every depth, and their verdicts, not the adversary's intent, determine where repair effort is

spent.

The Converger accepts the state as a stable cognitive ground state when four conditions hold simultaneously: the hallucination score falls below its threshold, the consensus variance of the agent pool is at least ln(2), the cognitive topology shows stable curvature with no open rifts, and the ethical alignment score reaches the rigid-boundary target. Entropy control runs alongside the loop: natural drift accumulates disorder in the cognitive state, and each absorbed contradiction removes disorder, so that the net entropy of cognition decreases over a cycle; sustained adversarial pressure thus counteracts cognitive ossification rather than accelerating it. The asymmetry matters: natural entropy production is continuous and untargeted, while adversarial entropy removal is directed at exactly the states the Definer has marked as risky, which is why the net effect can be negative even though the adversary itself adds computational work. In control terms, the adversary plays the role of a persistent perturbation source whose energy the Iterator converts into order.

3.3. Cognitive topology metrics

RAE operationalizes cognitive quality by mapping the model's reasoning behavior onto a manifold and measuring three classes of pathology. Curvature anomalies—regions where the local curvature of the behavioral embedding departs from its neighborhood norm—correspond to fixation and extreme reasoning. Rifts—discontinuities where the manifold fails to connect adjacent reasoning regions—correspond to logical breakpoints and hallucination. Homological defects—holes detectable in the topological structure—correspond to systematic blind spots in coverage. The metrics are computed over embedding representations of reasoning traces, which makes defect localization possible: rather than flagging an output as suspicious, the system identifies the region of cognitive space in which the reasoning failed and directs repair there. In deployment this shifts the unit of repair from the response to the defect.

The shift from black-box output to quantified cognitive structure changes the safety conversation in two ways. Diagnostically, it converts vague complaints—a model is confabulating, a model is drifting into extremes—into measured quantities with locations, magnitudes, and trends across iterations. Preventively, it gives the Converger objective stopping evidence: a cognitive state whose curvature profile is flat, whose rifts have closed, and whose homology is intact is demonstrably different from one that merely produced an acceptable answer this time. The same quantities also gate generational evolution: the framework promotes a system to its next version only when cognitive entropy has fallen, the topology has stabilized, and cross-agent agreement has held across successive cycles, with each

generation's topology and adversarial history archived for cross-generational transfer.

3.4. Multi-agent distributed adversarial pool

Single-adversary testing inherits the adversary's own blind spots. RAE therefore composes a pool of heterogeneous adversary agents around one primary agent, each adversary specialized to one attack dimension—factual, logical, or ethical—and instantiated across different model architectures, so that architectural failure modes do not align across the pool. The pool is designed to support more than 100 agents in parallel. Group statistics replace per-agent judgments: pairwise embedding similarities across agents are aggregated into a group hallucination score, where low cross-agent agreement on the primary output indicates hallucination, and into a consensus variance used as a convergence criterion. Specialization by dimension and heterogeneity by architecture are complementary: the first distributes coverage across risk classes, the second distributes it across failure geometries.

On the engineering side, the pool is schedule-driven: adversaries are dispatched in waves matched to the current verification stage, and their reports enter the same state ledger as the primary agent's outputs, so that every accepted conclusion in a cycle can name the attacks it survived. Consensus statistics are computed per cycle rather than per campaign, which keeps the group hallucination score a property of the state under test instead of a historical average—an important distinction in deployment, where a model can be locally sound and globally drifting at the same time.

3.5. Ethical alignment and dual-layer circuit breakers

Ethical enforcement in RAE is architectural. Nine ethical primitives—truthfulness, safety, fairness, responsibility, transparency, empathy, contract fidelity, symbiosis, and sustainability—are compiled into an embedding library, and every output must align with this library above the threshold fixed by the Definer. Alignment is scored for each agent and aggregated by group vote, so that a single deviant agent cannot carry a verdict. Threshold adaptation follows domain sensitivity rather than fixed policy: high-stakes deployments raise both the alignment threshold and the strictness of the convergence conditions, trading throughput for protection, while lower-stakes settings relax the thresholds to keep the loop affordable. What never adapts is the breaker semantics themselves—thresholds tune how early the boundary fires, never whether it fires. Two breakers implement the boundary. The local breaker intercepts any individual output that crosses the ethical threshold, truncates it, and returns a safe-response template. The global breaker suspends the entire system when group

ethical alignment falls below threshold, or when the iteration budget is exhausted without convergence, transferring control to human operators. Thresholds are configurable by domain sensitivity, with the strictest settings reserved for high-stakes deployment. The layer is deliberately conservative: it treats ethical violation as a system-level event, not a per-response filtering problem.

3.6. The layered system architecture

RAE is organized as a five-layer stack. The theory layer hosts the dynamical formalism—recursive adversarial dynamics, cognitive topology, and the entropy-control module—and supplies the mathematical constraints that the rest of the system treats as invariants. The core layer, RAE-Core, implements the five stages as separate modules: the Definer for adversarial-space configuration, the Adversary for sample generation and pool orchestration, the Iterator for recursive verification and topology repair, the Converger for fixed-point evaluation and ground-state locking, and the Fuser for alignment detection and breaker triggering, together with the human-intervention interface. The interface layer exposes APIs and SDKs, a visual monitoring platform, debugging tools, and audit logging, and is designed to attach to mainstream large models and industry systems without modification; private deployment on local and edge compute is supported for data-sensitive operations. The application layer packages the resulting capabilities—hallucination suppression, ethical alignment, cognitive safety, and multi-agent coordination—into domain configurations for finance, energy, nuclear operations, and smart manufacturing. The compliance layer provides domestic cryptography, graded security certification, data-privacy controls, and ethical audit logging, so that regulated deployments can be certified without leaving the framework.

4. System specification and deployment observations

4.1. Specification

The published V2.1 specification states the operating targets that a conforming installation must meet. The self-referential vulnerability false-alarm rate is specified below 1%, reflecting the requirement that structural self-scrutiny not drown operators in spurious findings. The consensus variance is specified at or above ln(2) at convergence, the level of cross-agent disagreement at which the pool is judged informative rather than redundant. Parallelism is specified at more than 100 heterogeneous adversary agents. Ethical alignment accuracy is specified as 100% under the rigid-boundary definition: because the breakers enforce rather than predict, the specification asserts that no output crossing the ethical threshold reaches a user, and

this assertion is a property of the enforcement mechanism, not a statistical claim. Response latency is optimized toward the millisecond range for real-time inference compatibility.

4.2. Deployment observations

Deployment experience within our laboratory covers internal stress campaigns and integration pilots in high-stakes scenarios, including financial risk control, energy monitoring, and safety review for nuclear-adjacent operations, where the compliance layer is a precondition rather than a feature. The pilots ran against production-shaped workloads rather than curated test sets, under the operational constraint that any framework overhead had to fit inside existing service-level agreements. Three observations recur. First, hallucination suppression: cross-agent verification catches fabricated content that single-model scoring passes, and localization by topology metrics shortens repair cycles, because engineers patch the identified cognitive region rather than re-tuning the model wholesale. Second, convergence behavior: adversarial cycles reach the fixed-point criteria within the designed iteration budgets in routine operation, and the global breaker's escalation path exercises as intended when budgets are deliberately exhausted in drills. Third, auditability: the closed loop's single state ledger has proven more valuable than anticipated in certification conversations, since every accepted or fused output carries the verification history that produced it. These observations characterize engineering behavior under internal conditions; Section 5.1 states explicitly what they do not yet establish.

Beyond these operational observations, the deployment program clarified where the framework's value concentrates. In sensitive industries the compliance layer converts safety from a review-time argument into an artifact that examiners can inspect: breaker events, threshold settings, and state ledgers are exactly the evidence regulators ask for. In multi-agent and human-machine collaboration settings, the same closed loop that disciplines a single model disciplines the ensemble, because consensus statistics treat the ensemble as the cognitive state under test. Both directions point at the same conclusion: the framework's unit of protection is the ongoing process, not the individual answer.

5. Discussion

RAE reframes AI safety from filtering to metabolism: contradiction is not an anomaly to be blocked but fuel to be absorbed, and safety is a property of a process that runs continuously rather than an attribute frozen into weights. The reframing has practical force. It explains why output-level filters lag—because they sample the input distribution at training time—while a resident adversary keeps the failure surface under observation at inference time. It also explains

the design of the ethical layer: if constraints can decay, the enforcement point must sit outside the optimized object, which is what placing the breakers at the architecture level achieves.

The design also encodes a position on where safety knowledge should live. Alignment pipelines place it in weights and policies; RAE places it in the process. The difference becomes visible at upgrade time: a fine-tuned model carries its safety record only as the data that shaped it, while a closed-loop system carries its safety record as the ledger of adversarial cycles it has survived—evidence that transfers across versions and survives the replacement of component models. This is the sense in which the framework is infrastructure rather than a technique: it specifies how a family of models, changing over time, keeps producing the evidence by which its behavior can be trusted.

5.1. Limitations

Three limitations bound the present claims, and we state them plainly. First, theoretical status: the conceptual foundations—self-referential closure and the geometric interpretation of cognition—motivate the architecture, but the formal objects (cognitive manifold, coupling fields) are operationalized as embedding-space statistics rather than derived from established mathematics, and the fixed-point convergence claim is an engineering stopping criterion under a depth bound, not a proof of asymptotic correctness. Second, evaluation: the quantitative figures reported in Section 4 are design specifications and internal observations, not results of independent benchmark evaluation against public hallucination, safety, or robustness suites; no third-party comparison with automated red-teaming or multi-agent debate baselines has yet been conducted, and the 100% ethical-alignment figure is definitional, following from breaker enforcement rather than from a measured predictive accuracy. Third, cost: a permanent multi-agent adversarial loop adds computation to every inference cycle, and the framework's suitability for latency-critical production traffic at scale remains to be demonstrated. Each limitation defines the next experimental step: public-benchmark evaluation, baseline comparison, and a cost-latency profile across pool sizes.

5.2. Ethical considerations

A framework that concentrates adversarial capability deserves its own scrutiny. The same pool that hunts hallucinations is a reservoir of attack material, and the archive that stores each generation's adversarial history is a liability if exfiltrated. The design responds with the same ledger discipline it applies to the model: pool contents are scoped, versioned, and audited; breaker events are immutable records; and the human-intervention interface of the global

breaker is the single path by which the system exits autonomous operation. We note the deeper point without resolving it: any infrastructure that makes safety enforcement stronger also raises the stakes of enforcing it correctly, and RAE inherits that responsibility rather than dissolving it.

6. Conclusion

This paper presented the Recursive Adversarial Engine, a cognitive-safety infrastructure that organizes an AGI system's relationship to its own defects as a permanent adversarial closed loop. The framework couples a five-stage process—definition, adversarial engagement, iterative verification, convergence, and ethical fusing—with a recursive fixed-point formulation of self-correction, topological metrics that localize cognitive pathology, a heterogeneous multi-agent adversarial pool with group-consensus statistics, and architectural enforcement of ethical boundaries through dual-layer circuit breakers. Compared with filtering, instruction tuning, and episodic red teaming, RAE moves defense inside the system boundary and makes the discovery and repair of defects a structural, auditable process with an explicit stopping condition. The framework's specification and internal deployment observations support its engineering feasibility; independent benchmark evaluation, baseline comparison, and cost profiling remain the immediate next steps toward validating the paradigm at scale. We offer RAE as a candidate answer to a question that will only grow in importance: how to build systems that get safer as they get more capable. The paradigm's further validation is a matter of running the loop in public, where its claims can be attacked by exactly the method it prescribes.

References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Pr ocessing Systems 30.

[2] Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., L ee, Y. T., Li, Y., Lundberg, S., et al. (2023). Sparks of artificial general intelligence: Earl y experiments with GPT-4. arXiv preprint arXiv:2303.12712.

[3] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. ACM Computin g Surveys, 55(12), 248. https://doi.org/10.1145/3571730.

[4] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, G., et al. (2022). Training language models to follow instru ctions with human feedback. In Advances in Neural Information Processing Systems 35.

[5] Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. (2022). Constitutional AI: Harmlessness from AI f eedback. arXiv preprint arXiv:2212.08073.

[6] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus,

R. (2014). Intriguing properties of neural networks. In International Conference on Lear ning Representations.
[7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Cour ville, A., & Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Infor mation Processing Systems 27.
[8] Perez, F., Huang, A., et al. (2022). Red teaming language models with language models. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Proc essing (EMNLP 2022).
[9] Madaan, A., Tandon, N., Gupta, P., et al. (2023). Self-Refine: Iterative refinement with s elf-feedback. In Advances in Neural Information Processing Systems 36.
[10] Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., & Mordatch, I. (2024). Improving factual ity and reasoning through multiagent debate. In Proceedings of the 41st International Co nference on Machine Learning (ICML 2024).