Verification Independence for Agentic AI

Designing Review Beyond Self-Correction and Multi-Agent Agreement

Johnny Kao

Independent Researcher, Tokyo, Japan

ORCID: https://orcid.org/0009-0004-1446-6001

Working Paper | Version 1.1 | 24 September 2026

Abstract

As agentic artificial intelligence systems increasingly produce, evaluate, and act on their own outputs, review architectures face a structural problem: adding a second judgment does not necessarily create an independent basis for judgment. This paper proposes verification independence as a task-level design property for agentic AI. Verification independence asks whether the review process is sufficiently separated from the production process to surface errors that the producer is systematically unlikely to detect. The framework distinguishes four dimensions of independence: framing, evidence, mechanism, and authority. It also formalizes a four-function review architecture consisting of a producer, challenger, evidence checker, and acceptance owner.

The framework is positioned relative to research on LLM self-correction, chain-of-verification, multi-agent debate, LLM-as-a-judge, correlated model errors, and emerging agentic verification systems. Existing work shows both the value and limits of additional model-based review: structured verification can improve outputs, while multi-agent agreement, self-critique, and LLM judging remain vulnerable to shared assumptions, correlated errors, evaluator bias, and noisy rubric verification. The contribution here is integrative rather than a claim that any individual verification technique is new. It treats independence as a property of the relationship between production and review, not as a synonym for reviewer count or model diversity.

The central claim is that a reviewer can provide another judgment without providing another independent basis for judgment. For consequential agentic workflows, reliable verification therefore requires deliberate separation of review roles and, where feasible, external evidence, deterministic tests, independent tools, or distinct authority boundaries. The paper further introduces a qualitative independence checklist, risk-adjusted verification depth, and safeguards against erosion of independence in long-horizon workflows.

Keywords: agentic AI; verification independence; independent verification; self-correction; multi-agent systems; LLM-as-a-judge; correlated errors; AI governance

Author note: This working paper develops and formalizes the verification architecture first introduced in Kao (2026c).

1. Introduction

Agentic AI changes verification from an occasional quality-control step into a recurring architectural function. An agent may research, write code, manipulate files, call external tools, prepare recommendations, or coordinate other agents. In each case, the system can produce work faster than a human can independently reconstruct it. This creates pressure to automate review as well as production.

A common response is to ask the producing model to critique itself, add a second model as a reviewer, run a multi-agent debate, or use an LLM-as-a-judge. These techniques can improve performance. Self-Refine, Chain-of-Verification, and structured self-correction all show that additional critique or verification can reduce errors under some conditions (Madaan et al., 2023; Dhuliawala et al., 2024; Wu et al., 2024). Multi-agent debate can also improve reasoning and factuality on selected tasks (Du et al., 2024). The important question, however, is not whether additional review is useful. It is whether the review provides an independent basis for detecting failure.

That distinction matters because production and review can fail together. The same model may carry its initial framing into self-review. Two agents may inherit the same mistaken assumption, source material, tool outputs, or objective. A judge may share preferences with the system it evaluates. Different models may still

exhibit correlated errors. In such cases, a workflow can contain multiple review steps while remaining structurally vulnerable to the same underlying mistake.

This paper proposes verification independence as a design property for agentic AI. The proposal does not equate independence with using a different model, provider, or human. Instead, it asks whether the review process has enough separation in framing, evidence, verification mechanism, and authority to expose errors that the producer is systematically unlikely to detect. The contribution is integrative: it connects model-level verification research with the older system-design principle that robust control should not require one actor to produce, validate, and finally accept the same consequential work.

2. Self-Review Is Not Independent Review

Self-correction research provides an important starting point because it demonstrates both the usefulness and limits of asking a model to inspect its own work. Madaan et al. (2023) report gains from iterative self-feedback across several tasks, while Huang et al. (2024) find that intrinsic self-correction can fail or even degrade reasoning when a model receives no external feedback. Kamoi et al. (2024), reviewing the broader literature, conclude that successful self-correction depends strongly on the source and reliability of feedback, with external feedback remaining particularly important outside tasks naturally suited to self-verification.

The contrast between these findings is not a contradiction that must be resolved by choosing one side. It suggests that self-review and independent verification answer different questions. A producer can sometimes notice and repair its own mistakes, especially when the task exposes checkable constraints. Wu et al. (2024), for example, improve self-correction by converting key conditions into verification questions. Dhuliawala et al. (2024) similarly reduce hallucination by having a model formulate verification questions and answer them independently from the initial draft. These methods are valuable precisely because they introduce verification structure rather than assuming that generic reflection is sufficient.

The architectural implication is narrower: successful self-correction does not establish independence. The same model can become a better reviewer of its own output without acquiring a genuinely separate evidentiary basis. When production and review share the same latent assumptions, omitted evidence, or misinterpreted task, self-review may reproduce the original blind spot. The relevant distinction is therefore not self-review versus no review, but internal review versus review with independent grounds.

3. More Reviewers Do Not Automatically Create Independence

Multi-agent systems appear to solve the dependence problem by adding additional models or roles. Evidence is mixed. Du et al. (2024) show that multi-agent debate can improve mathematical reasoning and factuality, while Smit et al. (2024) find that debate systems do not reliably outperform simpler strategies such as self-consistency or ensembling and can be sensitive to protocol choices. The broader lesson is not that debate fails, but that the existence of multiple agents is not itself a verification guarantee.

Empirical evidence on correlated model behavior sharpens this concern. Kim et al. (2025) evaluate more than 350 language models and find substantial correlation in model errors, including high correlation among capable models even when architectures or providers differ. Relatedness can also bias evaluation directly. Li et al. (2025) describe preference leakage in LLM-as-a-judge settings when generators and evaluators are the same model, inherit from one another, or belong to the same model family. These results make model count a weak proxy for independence.

LLM-as-a-judge remains useful because it can scale evaluation where human review is expensive. Zheng et al. (2023) show strong agreement between capable LLM judges and human preferences in open-ended evaluation. Yet judge systems also exhibit known biases. Wang et al. (2024) demonstrate positional bias, and recent agentic benchmarks find meaningful noise even among frontier judges when verifying complex rubrics (Peng et al., 2026). A second evaluator can therefore add signal without necessarily providing a distinct basis for judgment.

The problem can be summarized as correlated review failure: production and review can converge on the same incorrect acceptance decision because they depend on overlapping assumptions, evidence, mechanisms, or authority structures. This failure mode is especially important in agentic systems because outputs can immediately trigger external actions rather than remain advisory text.

Table 1. Review approaches and the verification-independence gap

Approach	Primary value	Independence limitation
Self-review / self-correction	Low-cost iterative critique and repair.	Producer and reviewer may share the same framing, model limitations, and omitted evidence.
Multi-agent debate / ensembles	Additional reasoning paths, disagreement, and candidate diversity.	Multiple agents can still share correlated errors, evidence, objectives, or tool outputs.
LLM-as-a-judge	Scalable rubric-based or preference evaluation.	Judge bias, preference leakage, position effects, and noisy rubric verification can affect acceptance.
External verifiers / evaluation probes	Checks claims or behavior against explicit criteria, trusted sources, or tests.	Independence depends on whether the external basis is itself controlled by or derived from the producer.

4. Verification Independence

Definition 1 (Verification Independence). Verification independence is a property of a review architecture in which the basis for accepting or rejecting an output is sufficiently separated from the process that produced the output to make correlated failure less likely. Independence can arise from differences in framing, evidence, verification mechanism, authority, or combinations of these dimensions.

This definition deliberately avoids treating independence as binary. A reviewer may be independent in one dimension and dependent in another. A different model can still use the producer's summary as its only evidence. A deterministic test can provide strong mechanism independence while remaining based on an incorrect acceptance criterion. A human reviewer can be organizationally independent while inheriting the same flawed source material. The design question is therefore where the review obtains a distinct basis for disagreement.

Table 2. Qualitative verification-independence checklist

Dimension	Diagnostic question	Weak independence	Stronger independence
Framing	Can the reviewer challenge the producer's assumptions and problem formulation?	Same instructions, assumptions, or producer-supplied reasoning.	Adversarial or independently specified framing; explicit search for counterexamples and missing assumptions.
Evidence	Can the reviewer reach evidence beyond the producer's selected context?	Relies on the same summaries, citations, retrieval results, or intermediate outputs.	Independent retrieval, primary documents, ground-truth databases, or source records.
Mechanism	Does verification use a checking mechanism distinct from generation?	Another free-form generative judgment using similar tools and criteria.	Deterministic tests, reconciliation, static analysis, formal checks, or specialized verifiers.
Authority	Can the producer control final acceptance or the verification process?	Producer can alter review conditions or approve its own consequential action.	Proposal, review, and release permissions are separated; acceptance belongs to a distinct owner or policy.

4.1 Correlated Review Failure

Definition 2 (Correlated Review Failure). A correlated review failure occurs when the production and review processes accept the same incorrect result because their errors arise from shared dependencies rather than independent evidence of correctness.

The definition is intentionally broader than model correlation. Shared failure can arise from the same retrieved source, the same task framing, the same hidden assumption, the same flawed acceptance criterion, or the same access permissions. Model diversity can reduce some forms of correlation, but it does not address every dependency. Conversely, the same model can sometimes participate in a more independent verification process when it is forced to answer independently derived checks against external evidence. Verification independence is therefore an architectural property, not an identity property.

5. A Four-Function Review Architecture

The verification architecture developed in Kao (2026c) separates review into four functions: producer, challenger, evidence checker, and acceptance owner. These functions need not correspond to four people or four models. The point is functional separation: different parts of the workflow answer different questions about correctness.

The producer creates the work. The challenger searches for hidden assumptions, contradictions, missing alternatives, and failure cases. The evidence checker verifies factual or behavioral claims against something outside the producer's own assertion. The acceptance owner decides whether the result satisfies the relevant standard for the consequence attached to it. In a coding workflow, for example, the producer may create a patch, a reviewer agent may challenge design assumptions, tests and scanners may provide external evidence, and a human or release policy may control production deployment.

Table 3. Four-function review architecture

Function	Primary question	Typical basis
Producer	What should be created or executed?	Task instructions, tools, context, and delegated authority.
Challenger	What could be wrong, missing, or misleading?	Alternative framing, counterexamples, adversarial review, independent reasoning path.
Evidence Checker	What can be verified outside the producer's opinion?	Primary sources, deterministic tests, databases, reconciliations, formal or programmatic checks.
Acceptance Owner	Is the work sufficient for the consequence attached to it?	Acceptance criteria, policy thresholds, residual risk, and final decision authority.

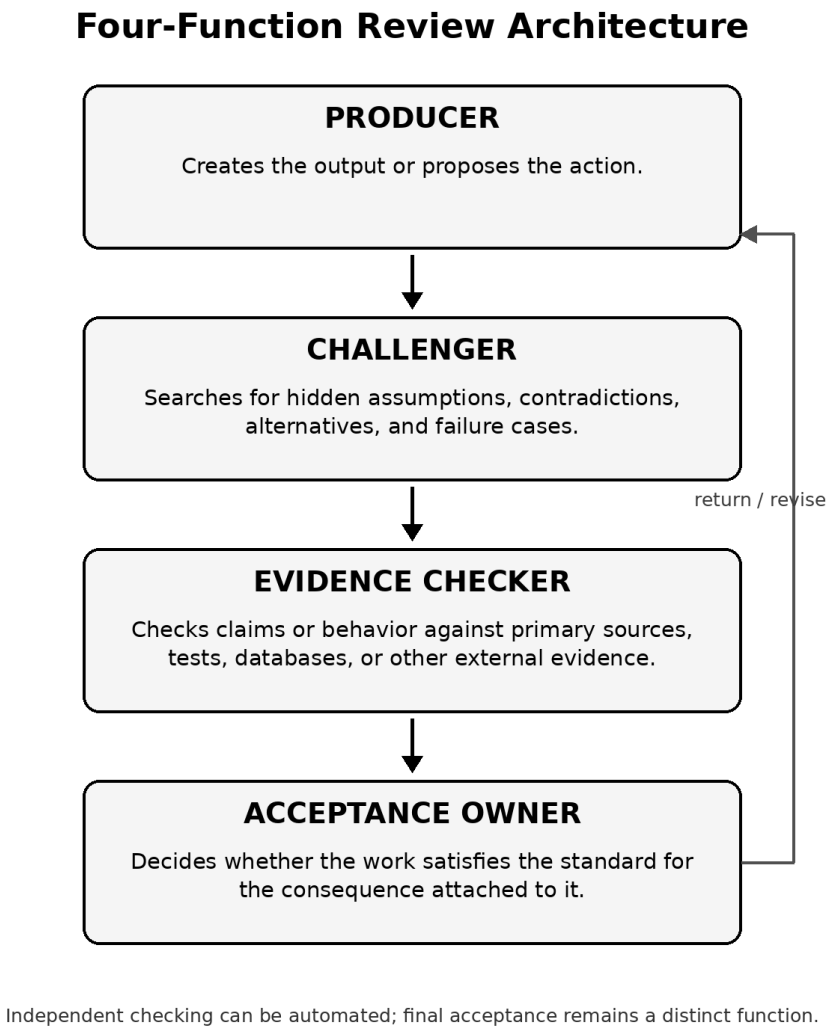

Figure 1. Simplified four-function review flow. Challenger and evidence-checking failures can return work to the producer; acceptance remains a distinct function.

The architecture is compatible with recent technical work rather than a replacement for it. NIST is developing evaluation probes that act as adversarial verifiers inside agentic workflows and compare claims against trusted corpora while producing structured audit trails (NIST, 2026). Yuan et al. (2026) similarly combine adversarial auditing with verifiable acceptance criteria for agent reasoning, while Kwok et al. (2026) treat verification as a distinct scaling axis and decompose criteria to improve verifier performance. These approaches reinforce the value of separating generation from explicit checking, even when both functions remain highly automated.

6. Externalized Verification

Principle 1 (Externalized Verification Principle). For consequential outputs or actions, the acceptance decision should, where feasible, depend on evidence, tests, or constraints that are not generated and controlled solely by the producer.

This principle does not require human verification. A deterministic test suite can be more independent than a second generative model when the question is whether software satisfies a specified behavior. A reconciliation can be more independent than an LLM reviewer when the question is whether financial totals balance. A source document can be more independent than a model critique when the question is whether a citation exists and supports a claim. The reviewer may still be an AI; the crucial question is whether the acceptance basis can disagree with the producer for reasons the producer does not control.

This distinction also clarifies the role of LLM verifiers. A model-based verifier can provide valuable signal, especially when criteria are decomposed and repeated evaluation reduces variance (Kwok et al., 2026). But the use of a verifier should not erase the need to inspect its dependencies. If the verifier is trained on related outputs, reads only the producer's evidence, or applies a rubric that does not capture the consequential failure mode, the architecture can remain dependent even when the verifier itself is accurate on benchmark tasks.

For agentic workflows, externalized verification is particularly important at points where the system changes external state. Before a high-consequence action is committed, review should preferentially draw on checkable evidence: policy rules, source records, environment state, permission boundaries, tests, or other observable constraints. The goal is not to eliminate judgment. It is to ensure that judgment is anchored to something more than agreement between generative systems.

6.1 Independence Erosion in Long-Horizon Workflows

Verification independence can erode during long-horizon work even when a reviewer begins from a separate role. As agents exchange summaries, shared memory, tool outputs, intermediate beliefs, and retrieved evidence, a reviewer can gradually inherit the producer's framing. The architecture may still contain two nominal roles while the informational basis for disagreement becomes increasingly similar.

Two lightweight safeguards are useful. Context isolation limits the amount of producer-generated reasoning and intermediate state that a reviewer receives when those details are not needed for verification. Independent evidence refresh requires the reviewer, at consequential checkpoints, to return to primary sources, current environment state, or independently retrieved evidence rather than relying only on the producer's summary. Neither safeguard guarantees correctness; both are intended to preserve a distinct basis for disagreement.

This is a design extension rather than a validated decay model. The relevant empirical question is whether particular forms of information sharing reduce detection of producer-originated errors over time, and whether selective isolation or evidence refresh restores useful independence without imposing excessive cost or latency.

7. Integration with Delegation and Attention-Constrained Oversight

Verification independence complements two related control layers. The delegation-contract framework specifies the task itself through objective, context, constraints, acceptance criteria, authority, and escalation (Kao, 2026a). The attention-constrained oversight framework then allocates human review according to risk, reversibility, authority, and the available human-attention budget (Kao, 2026b). Verification independence addresses a different question: when review occurs, does it provide an independent basis for deciding whether the work is acceptable?

The three layers can be read sequentially. Delegation defines what success and authority mean. Oversight determines which decisions deserve scarce human attention or stronger controls. Verification independence determines whether the checking process can detect producer-originated blind spots. In practical terms, acceptance criteria from the delegation contract should inform the evidence checker; escalation conditions determine when the acceptance owner must intervene; and high-consequence actions identified by the oversight framework justify stronger independence requirements.

7.1 Risk-Adjusted Verification Depth

Verification need not be equally independent for every action. The attention-constrained oversight framework already distinguishes actions by consequence; the same logic can determine how much separation is justified in the review architecture. As the consequence of correlated failure increases, verification should rely on progressively more independent evidence, mechanisms, and authority boundaries.

Table 4. Heuristic risk-adjusted verification depth

Consequence profile	Suggested verification depth	Illustrative pattern
Low / reversible	Lightweight independence	Same-model critique or simple automated checks; logging and later audit may be sufficient.
Moderate / bounded	Structured separation	Adversarial framing plus independent retrieval, specialized tools, or a separate checking mechanism.
High / hard to reverse	Strong independence	Independent evidence, deterministic or source-grounded verification, and separated acceptance authority before commitment.

These bands are heuristic rather than validated thresholds. Their purpose is to connect verification depth to consequence without implying a universal numeric independence score.

This integration also prevents an important category error. More human attention cannot compensate for structurally dependent evidence, just as a more independent reviewer cannot compensate for an undefined acceptance standard. Reliable control requires both a clear standard and a review process capable of challenging the system that produced the work.

8. Design Implications and Research Agenda

First, evaluation pipelines should document review dependencies, not only reviewer identities. The useful questions are whether the reviewer sees the producer's reasoning, whether it can access primary evidence, whether it uses a different checking mechanism, and whether it can block or alter the final action. These features are more informative than simply recording that two models participated.

Second, disagreement should be treated as information rather than automatically resolved by majority vote. Multi-agent agreement can improve robustness, but correlated error means consensus is not equivalent to correctness. A reviewer that disagrees for an evidence-backed reason may be more valuable than several reviewers that agree because they share the same premise.

Third, agentic verification should be tested against intentionally correlated failures. A useful empirical design would compare: (A) producer self-review; (B) a second instance of the same model with the same context; (C) a different model with the same context and evidence; (D) a reviewer with independently retrieved evidence; and (E) a mixed architecture that combines model review with deterministic or source-grounded checks. Outcomes could include false acceptance of seeded errors, error overlap, detection latency, cost, and reviewer disagreement. The objective would not be to produce a universal independence score, but to identify which forms of separation materially reduce correlated failure.

Fourth, high-stakes workflows should preserve acceptance ownership. Verification work can be delegated to models, tools, and automated probes, but the standard for release or commitment still requires an identifiable owner or policy. This is consistent with long-established separation-of-duties controls in information security, where access authorizations are designed so that critical functions are not concentrated in a single role (NIST, 2020).

Finally, future work should measure how verification independence changes across long-horizon agent trajectories. Experiments could vary the amount of shared memory, producer summaries, retrieval overlap, and tool-output reuse while measuring correlated false acceptance. They could then test whether context isolation or independent evidence refresh preserves error-detection diversity, and at what additional latency or cost.

9. Limitations

The framework is conceptual and does not provide a validated quantitative measure of verification independence. The four dimensions identify design dependencies, but their relative importance will vary by domain. A deterministic test may dominate mechanism independence in software while evidence provenance may matter more in research or legal work. The risk-adjusted verification-depth bands are likewise heuristic and should not be interpreted as calibrated safety thresholds.

Independence also does not guarantee correctness. A reviewer can be independent and still wrong, incomplete, or poorly calibrated. The framework addresses correlated failure, not every source of verification error. Likewise, external evidence can itself be stale, manipulated, or insufficient.

The four-function architecture simplifies real organizations and agent systems. In practice, roles may be distributed across multiple tools, people, policies, and services, and the same component may perform different functions at different times. Empirical work is needed to determine when separation improves outcomes enough to justify additional cost and latency.

Finally, some recent verification results cited here are preprints or ongoing technical programs. They are included because agentic verification is evolving rapidly, but their claims should be interpreted according to publication status rather than treated as settled evidence.

10. Conclusion

Agentic AI makes automated review increasingly necessary, but review count and verification independence are not the same thing. Self-correction can improve outputs. Multi-agent debate can add useful reasoning paths. LLM judges and verifiers can scale evaluation. None of these mechanisms, by themselves, ensures that the review process has a distinct basis for detecting the producer's errors.

Verification independence reframes the design problem around four forms of separation: framing, evidence, mechanism, and authority. The associated four-function architecture separates production, challenge, evidence checking, and acceptance ownership without requiring every function to be human. Its central principle is that consequential acceptance should, where feasible, depend on evidence or checks that the producer does not fully control.

The central claim is simple: adding a reviewer does not create independent verification when the reviewer inherits the producer's assumptions, evidence, tools, or authority. A second model can provide another judgment without providing another independent basis for judgment. Verification independence must therefore be designed.

AI Use Disclosure

OpenAI's ChatGPT was used only for literature discovery and bibliographic verification. The verification-independence framework and underlying concepts originate in the author's prior published work (Kao, 2026c). The author independently reviewed all cited sources and remains solely responsible for the final manuscript, its arguments, source selection, and citations.

References

Dhuliawala, S., Komeili, M., Xu, J., Raileanu, R., Li, X., Celikyilmaz, A., & Weston, J. (2024). Chain-of-Verification reduces hallucination in large language models. Findings of the Association for Computational Linguistics: ACL 2024, 3563-3578. https://doi.org/10.18653/v1/2024.findings-acl.212

Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., & Mordatch, I. (2024). Improving factuality and reasoning in language models through multiagent debate. Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 11733-11763. https://proceedings.mlr.press/v235/du24e.html

Huang, J., Chen, X., Mishra, S., Zheng, H. S., Yu, A. W., Song, X., & Zhou, D. (2024). Large language models cannot self-correct reasoning yet. International Conference on Learning Representations. https://proceedings.iclr.cc/paper_files/paper/2024/hash/8b4add8b0aa8749d80a34ca5d941c355-Abstract-Conference.html

Kamoi, R., Zhang, Y., Zhang, N., Han, J., & Zhang, R. (2024). When can LLMs actually correct their own mistakes? A critical survey of self-correction of LLMs. Transactions of the Association for Computational Linguistics, 12, 1417-1440. https://doi.org/10.1162/tacl_a_00713

Kao, J. (2026a). Delegation contracts for agentic AI: A task-level framework for objectives, context, constraints, acceptance criteria, authority, and escalation. Working paper.

Kao, J. (2026b). Attention-constrained oversight for agentic AI: Risk, reversibility, authority, and human attention. Working paper.

Kao, J. (2026c). How to stay in control when AI does the work: A practical framework for delegation, judgment, and accountability. Independent publication.

Kim, E. M., Garg, A., Peng, K., & Garg, N. (2025). Correlated errors in large language models. Proceedings of the 42nd International Conference on Machine Learning, PMLR 267, 30038-30066. https://proceedings.mlr.press/v267/kim25e.html

Kwok, J., Li, S., Atreya, P., Liu, Y., Jiang, Y., Finn, C., Pavone, M., Stoica, I., & Mirhoseini, A. (2026). LLM-as-a-Verifier: A general-purpose verification framework [Preprint]. arXiv. https://arxiv.org/abs/2607.05391

Li, D., Sun, R., Huang, Y., Zhong, M., Jiang, B., Han, J., Zhang, X., Wang, W., & Liu, H. (2025). Preference leakage: A contamination problem in LLM-as-a-judge [Preprint]. arXiv. https://arxiv.org/abs/2502.01534

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B. P., Hermann, K., Welleck, S., Yazdanbakhsh, A., & Clark, P. (2023). Self-Refine: Iterative refinement with self-feedback. Advances in Neural Information Processing Systems, 36. https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html

National Institute of Standards and Technology. (2020). Security and privacy controls for information systems and organizations (NIST SP 800-53 Rev. 5). https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final

National Institute of Standards and Technology. (2026). Building evaluation probes into agentic AI. https://www.nist.gov/programs-projects/building-evaluation-probes-agentic-ai

Peng, Y., Qi, Y., Peng, H., Xia, H., He, G., Shi, X., Xuan, R., Lu, S., Liu, Y., Hu, Z., Liu, Y., Hou, L., Xu, B., & Li, J. (2026). Can LLM-as-a-Judge reliably verify rubrics in agentic scenarios? [Preprint]. arXiv. https://arxiv.org/abs/2606.29920

Smit, A. P., Grinsztajn, N., Duckworth, P., Barrett, T. D., & Pretorius, A. (2024). Should we be going MAD? A look at multi-agent debate strategies for LLMs. Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 45883-45905. https://proceedings.mlr.press/v235/smit24a.html

Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., & Sui, Z. (2024). Large language models are not fair evaluators. Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics, 9440-9450. https://doi.org/10.18653/v1/2024.acl-long.511

Wu, Z., Zeng, Q., Zhang, Z., Tan, Z., Shen, C., & Jiang, M. (2024). Large language models can self-correct with key condition verification. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 12846-12867. https://doi.org/10.18653/v1/2024.emnlp-main.714

Yuan, W., Lin, C., Chen, J., Xu, J., Wang, X., & Ngai, E. C.-H. (2026). Verify before you commit: Towards faithful reasoning in LLM agents via self-auditing. Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics, 31201-31225. https://doi.org/10.18653/v1/2026.acl-long.1440

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. Advances in Neural Information Processing Systems, 36. https://papers.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html