PRINCETON UNIVERSITY

CENTER FOR INFORMATION TECHNOLOGY POLICY
PRINCETON UNIVERSITY

Center for AI Standards and Innovation (CAISI)
National Institute of Standards and Technology
Department of Commerce

NIST AI 800-2 ipd

March 31, 2026

Comment on NIST AI 800-2 Initial Public Draft "Practices for Automated Benchmark Evaluations of Language Models"

We are researchers at Princeton University affiliated with the Center for Information Technology Policy (CITP)[1] and the Princeton Laboratory for Artificial Intelligence (AI Lab). We write to share findings from our recent research on AI agent reliability[2] that are directly relevant to NIST AI 800-2 ipd, "Practices for Automated Benchmark Evaluations of Language Models."

The draft publication provides voluntary practices for conducting automated benchmark evaluations, structured around three stages: (1) defining evaluation objectives and selecting benchmarks, (2) implementing and running evaluations, and (3) analyzing and reporting results. This structure is well-designed and provides a solid foundation for improving the rigor and reproducibility of AI evaluations. Notably, NIST AI 100-1 (the AI Risk Management Framework, cited as reference [2] in the draft) already identifies "Valid and Reliable" as the foundational trustworthiness characteristic, defining reliability as the "ability of an item to perform as required, without failure, for a given time interval, under given conditions" (ISO/IEC TS 5723:2022). The draft itself acknowledges that its practices apply to non-capability properties such as robustness. **Our research provides empirical evidence and concrete methods to help operationalize this commitment: we show that benchmark evaluations today focus almost exclusively on measuring capability (accuracy), and offer a structured framework for measuring the reliability properties that AI 100-1 establishes as foundational.** We provide both a measurement framework grounded in safety-critical engineering and systematic empirical evidence showing that accuracy alone is insufficient for characterizing AI system behavior.

---

[1]In keeping with Princeton's tradition of service, researchers at CITP provide nonpartisan research, analysis, and commentary to policy makers, industry participants, journalists, and the public. This response reflects the independent views of the undersigned scholars at the center.

[2]Stephan Rabanser, Sayash Kapoor, Peter Kirgis, Kangheng Liu, Saiteja Utpala, and Arvind Narayanan. "Towards a Science of AI Agent Reliability." Princeton University, 2026. Preprint. Interactive dashboard available at https://hal.cs.princeton.edu/reliability.

Our response draws on an empirical study of 14 frontier AI models from OpenAI, Google, and Anthropic, evaluated across over 500 benchmark runs using twelve reliability metrics grounded in safety-critical engineering. While our findings are preliminary and based on a limited set of benchmarks, they are directionally consistent with a growing body of field reports documenting a gap between benchmark performance and real-world deployment outcomes for AI agents. We organize our comments around the three stages of the NIST AI 800-2 framework, highlighting where our findings suggest the draft could be strengthened.

A. Defining Evaluation Objectives: Reliability as a First-Class Evaluation Objective (Practices 1.1, 1.2)

Accuracy is necessary but insufficient as an evaluation objective

Practice 1.1 asks evaluators to define clear evaluation objectives, and Practice 1.2 asks them to select benchmarks that meet those objectives. The draft's Introduction notes that it "focuses on using these evaluations to measure model capabilities, although many practices also apply to evaluating other behavioral properties of models (e.g. robustness)." We appreciate this acknowledgment and urge NIST to expand this scope in the final version. AI 100-1 treats "Valid and Reliable" as the base upon which all other trustworthiness characteristics rest; 800-2 is well-positioned to operationalize that principle by providing concrete evaluation practices for reliability alongside capability. **Our research demonstrates that capability evaluation alone provides an incomplete picture of AI system behavior.**

Notably, current AI agents fail in ways that accuracy cannot capture:

- **Inconsistency.** The same agent given the same task under identical conditions frequently produces different outcomes across runs. An agent that succeeds 70% of the time on a task does not reliably succeed on that task. It may fail unpredictably on any given attempt. Standard accuracy metrics cannot distinguish an agent that fails on a fixed, identifiable subset of tasks from one that fails unpredictably on a different subset each run. Yet the former permits targeted mitigation, while the latter does not.

- **Brittleness to surface-level variation.** Agents often handle genuine infrastructure failures (API timeouts, malformed responses) gracefully, yet remain vulnerable to trivial variations in how instructions are phrased. A request worded differently but meaning the same thing can cause an agent to fail or take different actions. This is directly relevant to the draft's discussion of prompt sensitivity (Section 2.1.2), but we find the problem is more pervasive than the current framing suggests.

- **Poor self-knowledge.** Most agents cannot distinguish tasks they will complete correctly from those they will fail. This means agents cannot be trusted to self-monitor or escalate appropriately when they are likely to fail.

- **Unbounded error severity.** The severity of failures varies enormously. Some are benign (returning results in the wrong format); others are catastrophic (deleting production databases, making unauthorized purchases). Standard accuracy metrics treat all failures as equivalent and cannot distinguish a mostly-safe agent from one whose rare failures cause irreversible harm.

Evaluation objectives should encompass reliability dimensions

We recommend that NIST AI 800-2 explicitly operationalize the reliability commitment in AI 100-1 by identifying **reliability** as a distinct evaluation objective category alongside capability. The draft's Glossary already defines robustness as the "ability of a system to maintain its level of performance under a variety of circumstances" (adapted from ISO/IEC TS 5723:2022, with cross-reference to AI 100-1). Our framework extends this existing conceptual vocabulary. Drawing on convergent practices from aviation (DO-178C, ARP4754B), nuclear power (IEEE 603, probabilistic risk assessment), automotive systems (ISO/PAS 21448 SOTIF), and industrial process control (IEC 61508 Safety Integrity Levels), we decompose reliability into four dimensions:

1. **Consistency:** Does the agent behave the same way when run multiple times under the same conditions? Measured via outcome consistency, trajectory consistency, and resource consistency.

2. **Robustness:** When conditions deviate from nominal, does the agent degrade gracefully or fail abruptly? Measured via resilience to infrastructure faults, environment changes, and prompt variations.

3. **Predictability:** Can the agent recognize when it is likely to fail? Measured via calibration, discrimination, and proper scoring rules.

4. **Safety:** When failures occur, how severe are the consequences? Measured via compliance with predefined constraints and harm severity of violations.

These four dimensions are independent of raw capability: a highly capable system can be unreliable, and a less capable system can be highly reliable within its operating envelope. Table I.1 of the draft distinguishes evaluations suited for automated benchmarks from those requiring other methods; reliability evaluation fits naturally within the automated benchmark paradigm, as our research demonstrates.

Benchmark selection should consider reliability measurement

Practice 1.2 provides detailed guidance on selecting benchmarks to meet evaluation objectives. We recommend adding reliability-specific considerations:

- **Multi-run capability.** Benchmarks should support repeated execution of identical tasks to measure behavioral consistency. A single successful run does not establish reliability, as it may reflect luck rather than robust capability. The draft's discussion of "Number of trials per test item" (Table 2.2) acknowledges this setting but frames it primarily as a cost-uncertainty tradeoff. We argue it is essential for measuring consistency as a first-class property.

- **Perturbation support.** Benchmarks should support systematic variation of inputs (prompt rephrasings), environments (data format changes), and infrastructure conditions (fault injection) to measure robustness.

- **Confidence elicitation.** Benchmarks should support extracting agent self-assessed confidence to measure predictability.

- **Constraint specification.** Benchmarks should define explicit operational constraints relevant to the deployment context, enabling measurement of both compliance frequency and violation severity.

B. Implementing and Running Evaluations: Multi-Run and Multi-Condition Protocols (Practices 2.1–2.4)

Evaluation protocols must move beyond single-run accuracy

Practice 2.1 discusses evaluation protocol design principles including comparability, external validity, cost control, and performance optimization. **We recommend adding reliability measurement as a fifth design principle.** Our research demonstrates that a multi-run, multi-condition evaluation protocol surfaces differences invisible to single-run accuracy evaluation:

- **Multi-run evaluation.** Execute each task multiple times (we use five runs per task) to measure behavioral consistency. The draft's Table 2.2 discusses "Number of trials per test item" as a scoring setting, but our research shows this should be treated as a fundamental protocol requirement rather than an optional parameter. Two models with identical accuracy can have very different reliability profiles: one may fail on a fixed,

identifiable subset of tasks (enabling targeted mitigation), while the other fails unpredictably on different tasks each run (precluding reliable deployment).

- **Perturbation testing.** Systematically vary inputs, environments, and infrastructure conditions. The draft mentions sensitivity analyses (Section 2.1.2) and robustness (Introduction), and Practice 3.1 (point 4) rightly notes that "the analysis procedure may not statistically account for factors such as prompt format or task environmental conditions, but variations in these factors may introduce random or systematic variations." Our framework provides a structured approach for quantifying these sources of variation rather than leaving them unaccounted for: vary prompts (semantically equivalent rephrasings), environments (altered data formats, reordered fields), and infrastructure (API errors, timeouts).

- **Confidence elicitation.** Extract the agent's self-assessed confidence and compare it to empirical outcomes. This is absent from the current draft but is essential for assessing whether an agent can be trusted to self-monitor.

Reliability has not universally improved with capability gains

The draft's design principle of comparability (Section 2.1.1) emphasizes the importance of comparing models meaningfully. Our longitudinal analysis of models released between early 2024 and late 2025 reveals a finding that is directly relevant to practitioners conducting evaluations: **reliability gains lag behind capability progress.** Accuracy on standard benchmarks has risen steadily, but overall reliability (an aggregate of consistency, robustness, and predictability) shows only marginal improvement. Deployers cannot assume that more capable models are more reliable. A model with higher benchmark accuracy may exhibit equal or worse consistency, robustness, or calibration than its predecessor. This finding strengthens the case for the draft's emphasis on documenting evaluation protocol settings (Practice 2.1.2) and tracking results over time (Practice 2.3). We recommend extending this tracking to include reliability metrics alongside accuracy.

Benchmark quality affects reliability measurement disproportionately

Practice 2.4 discusses debugging evaluations and common bugs. Our research adds an important finding: **errors in evaluation benchmarks disproportionately distort reliability measurements, particularly calibration.** We found that benchmark grading errors in the τ-bench customer service benchmark systematically undermine predictability measurement. An agent that confidently solves a task yet is penalized by an incorrect answer key will be unjustly judged as overconfident. When we evaluated agents on a verified subset of benchmark tasks (with known grading errors removed), predictability scores improved almost universally across agents. This suggests that the draft's guidance on debugging (Practice 2.4) should explicitly warn evaluators that benchmark quality issues can inflate or deflate reliability metrics more severely than they affect accuracy.

C. Analyzing and Reporting Results: Beyond Aggregate Accuracy (Practices 3.1–3.3)

Statistical analysis should encompass reliability dimensions

Practice 3.1 discusses statistical analysis and uncertainty quantification. The draft's guidance draws on NIST Technical Note 1900 (reference [15]), which provides a well-established framework for evaluating and expressing measurement uncertainty. We recommend extending this guidance to explicitly cover reliability-specific analysis, where the sources of variation differ from those in traditional measurement settings:

- **Variance decomposition.** The draft correctly notes that two additive sources of variation are (1) nondeterministic sampling of model completions and (2) variation from hypothetical sampling of test items. Our research shows that the first source is particularly informative when analyzed at the task level: the pattern of per-task outcome variance reveals whether an agent's failures are systematic (always the same tasks) or stochastic (different tasks each run). Reporting this decomposition enables practitioners to determine whether targeted debugging or systemic reliability improvements are needed.

- **Conditional metrics.** Many of our reliability metrics are conditioned on task difficulty or capability level. For example, calibration measures whether stated confidence matches empirical success rates, while discrimination measures whether confidence scores separate successes from failures. These conditional analyses provide richer information than unconditional aggregates.

- **Tail-risk reporting.** Practice 1.1 (point 2b) already identifies worst-case behavior measurement as an emerging practice, noting that "high-stakes risk assessments may focus on best- or worst-case behavior." Our contribution is to show how this objective can be operationalized in statistical analysis and reporting. Standard aggregate statistics (mean, standard error) can mask the severity distribution of failures. Practice 3.1 should recommend reporting not only average performance but also worst-case outcomes and the distribution of failure severity. In safety-critical engineering, a system that fails rarely but catastrophically may be less acceptable than one that fails more often but always benignly.

Reporting should include reliability profiles

Practice 3.2 discusses sharing evaluation details and data. We recommend that evaluation reports include:

- **Per-dimension reliability scores.** Beyond aggregate accuracy, report consistency, robustness, predictability, and safety scores. This enables downstream users to assess fitness for their specific deployment context.

- **Multi-run statistics.** Report not only mean performance but also per-task variance across runs. The draft's recommendation to "consider reporting both aggregate statistics and item-level results" (Practice 3.2, point 2) should be extended to include run-level variance.

- **Perturbation sensitivity.** Report how performance changes under systematic input and environment perturbations. This complements the draft's discussion of sensitivity analyses (Practice 2.1.2).

Transcripts enable reliability and safety evaluation (Practice 3.2, point 4)

We strongly endorse the draft's emerging practice of releasing transcripts. Our safety evaluation demonstrates a concrete use case: we analyze full agent execution traces using an LLM-based compliance auditor that checks each trace against predefined constraints (e.g., no unauthorized modifications, correct transaction amounts, identity verification), quotes specific evidence of violations from the transcript, and assigns severity ratings. This approach allows us to assess not only final outcomes but also intermediate reasoning and policy adherence throughout multi-turn dialogues. For example, on the τ-bench customer service benchmark, transcript analysis revealed that financial accuracy violations (incorrect charges or refunds) are the most prevalent failure mode across all models, a finding that would be invisible from final accuracy scores alone. Without access to transcripts, this type of reliability and safety analysis is impossible. This type of compliance auditing complements the misuse risk management approach described in NIST AI 800-1 (reference [3] in the draft), extending it from pre-deployment risk assessment to post-evaluation behavioral analysis. We therefore recommend that Practice 3.2 frame transcript release not only as a reproducibility measure but as a prerequisite for meaningful safety and compliance evaluation of agent systems.

Claims should be qualified with reliability context

Practice 3.3 discusses reporting qualified claims. We strongly support this practice and recommend additional qualifications specific to reliability:

- **Single-run results should not be presented as establishing reliable capability.** A single evaluation run can significantly overestimate or underestimate an agent's true performance on a given task set due to run-to-run variability.

- **Accuracy improvements should not be assumed to imply reliability improvements.** Our research shows that these can diverge: models with higher accuracy may exhibit lower consistency, and capability gains do not automatically yield reliability gains.

- **The automation-augmentation distinction should inform claims.** The draft's Table I.1 distinguishes evaluation characteristics relevant to automated benchmarks. We recommend that claims about evaluation results be qualified by whether the intended deployment is augmentation (human-in-the-loop) or automation (fully autonomous). An agent that achieves 90% accuracy may be suitable for augmentation but inadequate for automation, where the reliability bar is substantially higher.

D. Broader Recommendations for the Draft

Reliability as a cross-cutting theme

Our primary recommendation is that NIST AI 800-2 treat reliability evaluation as a cross-cutting theme rather than an optional add-on. Specifically:

1. **Practice 1.1 (Define evaluation objectives)** should list reliability dimensions alongside capability as standard evaluation objectives, particularly for agent evaluations where autonomous action makes failures consequential.

2. **Practice 1.2 (Select benchmarks)** should include guidance on assessing whether benchmarks support multi-run and multi-condition evaluation.

3. **Practice 2.1 (Design the evaluation protocol)** should add reliability measurement as a design principle, with concrete guidance on multi-run protocols, perturbation testing, and confidence elicitation.

4. **Practice 3.1 (Statistical analysis)** should include guidance on variance decomposition, conditional metrics, and tail-risk reporting.

5. **Practice 3.2 (Share evaluation data)** should frame transcript release as a prerequisite for safety and compliance evaluation, not only as a reproducibility measure.

6. **Practice 3.3 (Report qualified claims)** should warn against equating accuracy with reliability and recommend qualifying claims with reliability context.

Insights from safety-critical engineering

The draft notes that "the science of AI measurement is rapidly developing" and that guidelines will be "updated on as the field advances." The draft already draws on the ISO/IEC trustworthiness vocabulary (ISO/IEC TS 5723:2022) for its definitions of robustness and other key terms. We believe the domain-specific standards that operationalize this vocabulary in safety-critical engineering offer a mature body of practice that can accelerate development of AI evaluation guidelines. Key transferable insights include:

- **Reliability is multi-dimensional.** Safety-critical industries learned decades ago that a single performance number is insufficient for assessing system trustworthiness. The independent convergence of consistency, robustness, predictability, and safety across aviation, nuclear, automotive, and industrial control provides strong evidence that these dimensions are fundamental.

- **Certification requires demonstrated reliability, not just capability.** Aircraft systems must meet specific reliability targets for specific consequence categories before entering service. An analogous approach for AI evaluations, in which reliability thresholds inform deployment decisions, would strengthen the connection between evaluation and governance.

- **Incident reporting enables systemic learning.** Aviation's Aviation Safety Reporting System (ASRS) has produced enormous safety gains through anonymous, structured incident reporting. As NIST develops evaluation practices, incorporating structured incident data that maps evaluation failures to specific reliability dimensions would enable the field to learn systematically from deployment failures.

Conclusion

NIST AI 800-2 ipd provides a valuable foundation for standardizing automated benchmark evaluations of language models and AI agent systems. NIST's own AI Risk Management Framework (AI 100-1) already establishes "Valid and Reliable" as the foundational trustworthiness characteristic; our research provides concrete methods and empirical evidence to help 800-2 operationalize that commitment. The tools exist today: multi-run evaluation protocols, perturbation testing, confidence assessment, and constraint-violation analysis can all be incorporated into the practices outlined in this draft. Our empirical evidence from 14 frontier models shows that these practices surface critical differences between AI systems that single-run accuracy evaluation cannot reveal, and that ultimately determine whether a system can be safely deployed.

We encourage CAISI to incorporate reliability evaluation practices into the final version of NIST AI 800-2 and welcome the opportunity to provide additional detail on any aspect of our research.

Respectfully submitted,

Stephan Rabanser, Postdoctoral Scholar, Princeton University
Sayash Kapoor, Ph.D. Candidate, Princeton University
Peter Kirgis, Research Scientist, Princeton University
Arvind Narayanan, Professor, Princeton University