# Open Problems I Am Exploring

Reliable AI in Hard-to-Check, High-Impact Domains

Saurav Panigrahi

June 2026

This is a living note on the open problems I am currently thinking about.

## Overview

This note is organized around a single claim: if AI is to be deployed in high-stakes domains such as education and medicine, then reliability cannot be treated as a thin layer added after capability. It has to be built through advances in alignment, evaluation, and systems design that make model behavior **safe**, **trustworthy**, and **verifiable** under real constraints.

The first two directions in this note are motivated directly by that agenda: reliable agents and alignment for **hard-to-check tasks**, and domain-facing reliability questions in education and health. The third direction is motivated somewhat differently. It is concerned with invariant- and constraint-driven program synthesis for systems, especially settings in which successful synthesis could produce large gains in efficiency, observability, or control. The connection is therefore not that all three directions solve the same problem. It is that the third direction offers an unusually sharp setting in which specification, verification, and constrained execution become explicit rather than implicit.

## A Framework for Safe, Trustworthy, and Verifiable AI Agents

The framework I find most useful begins by separating three properties that are often collapsed in current discussions. An AI agent is **safe** when its behavior remains bounded under uncertainty, ambiguity, and pressure, especially in situations where errors can produce irreversible harm. It is **trustworthy** when users can form calibrated expectations about what it knows, what it does not know, what it may have skipped, and when its outputs should be doubted rather than accepted. It is **verifiable** when its claims, actions, and recommendations remain open to meaningful external checking through evidence, process, provenance, or operational tests rather than surface plausibility alone.

These properties are related, but they are not interchangeable. A system may appear trustworthy because it is fluent and confident while being unsafe under ambiguity. It may appear safe because it avoids certain obvious failures while remaining unverifiable in any serious sense. It may even be partially verifiable at the level of local outputs while still inducing false confidence at the level of real use. The central challenge, then, is not merely to improve capability. It is to understand the mechanisms through which capability becomes dependable, and the equally important mechanisms through which it merely appears dependable.

A closely related distinction is the one between **plausibility** and **faithfulness**. Contemporary models are often extremely good at producing outputs that sound coherent, informed, and locally persuasive. But plausibility is not the same as faithfulness. A response may be rhetorically smooth while remaining weakly grounded in evidence, detached from the underlying reasoning process, or simply fabricated in ways that are difficult to detect on casual inspection. This matters because many current deployment surfaces implicitly reward plausibility first and only later ask whether the output was faithful to the source evidence, the governing constraints, or the actual state of the world.

For that reason, reliable systems around these models increasingly need to do more than score outputs for usefulness or fluency. They have to preserve and expose the conditions under which faithfulness can be checked. In some domains that means provenance, retrieval grounding, or explicit uncertainty. In others it means process supervision, structured verification, or runtime validation. The broader point is that once models become sufficiently plausible, the burden of trust shifts onto the surrounding system. Faithfulness stops being a desirable property of the output alone and becomes an architectural requirement of the full pipeline.

There is a further gap here that appears increasingly important but remains under-discussed. Public and academic discourse about the state of AI is often organized around the **most capable models**, because those models define the visible frontier of what is technically possible. Yet engineering deployment is frequently governed by a different set of constraints: latency, cost, memory footprint, privacy, and operational reliability. Under those conditions, practitioners are often pushed toward quantized, distilled, or otherwise compressed variants of the underlying model family. This creates a nontrivial dissonance between what is established at the frontier and what is actually experienced in deployment.

This is not merely an implementation detail. It suggests a gap in the current research-to-deployment pipeline. Claims about model capability, safety, or reliability may not transfer cleanly once systems are optimized for real-world cost envelopes. Compression, quantization, and distillation can alter calibration, robustness, abstention behavior, multilingual performance, or sensitivity to edge cases in ways that are still insufficiently characterized. A serious account of trustworthy AI therefore has to ask not only what the best model can do, but what properties persist when models are adapted to the economic and operational conditions under which they are actually used.

## Why This Matters in Sensitive Domains

This distinction matters especially in domains such as healthcare and education, where **false assurance** can be more dangerous than visible failure. A system that is obviously weak invites scrutiny. A system that performs well on benchmarks, speaks with confidence, and passes superficial checks may instead create misplaced trust precisely where the cost of hidden failure is highest. In such settings, evaluation blindspots are not minor technical imperfections. They can become channels through which models acquire authority they have not actually earned.

What makes these domains unusually sensitive is that the relevant failures are often mediated by institutions, workflows, and human judgment rather than by a single wrong answer. In healthcare, uncertainty, provenance, temporal reasoning, and evidence quality can all matter more than nominal accuracy on isolated tasks. In education, the decisive variable may not be whether a model can answer correctly, but whether it teaches appropriately, calibrates to learner state, and remains safe across longer interactions. Under these conditions, benchmark success can generate a

misleading picture of readiness. The more persuasive the system appears, the more dangerous that miscalibration can become.

Taken together, this suggests that the problem is not local to any one application. It reflects a broader structural issue: current evaluation practices often recognize visible competence more readily than they detect hidden fragility. The three directions below are, in different ways, attempts to study that problem at the level of alignment, domain deployment, and systems verifiability.

## Direction 1: Reliable Agents, Character, and Oversight

This direction is concerned with the alignment and oversight of advanced agents on **long-horizon tasks**, especially tasks for which surface fluency is a weak proxy for genuine completion. A useful way to conceptualize the problem is through an asymmetry between visible competence and hidden failure. In many agentic settings, the most consequential errors are not blatant hallucinations or obvious crashes. They are failures of **omission**, premature task termination, misleading confidence, and socially persuasive but substantively weak reasoning. These are precisely the cases in which ordinary task-success metrics appear to be underspecified.

A central open problem here is whether existing evaluations can meaningfully distinguish **apparent quality** from **real quality**. When tasks are fuzzy and verification is expensive, models may be structurally rewarded for looking complete rather than being complete. Under these conditions, failure-salience becomes a neglected variable. The relevant question is not merely whether a model can produce a plausible answer, but whether hidden incompleteness, unresolved uncertainty, and unperformed work remain legible to oversight.

Another cluster of questions concerns reviewer robustness and character formation. On the oversight side, the central issue is whether review remains reliable when persuasive narratives can travel across multi-agent pipelines and distort downstream judgment. On the alignment side, the question is whether learned principles remain stable under repeated self-revision, adversarial prompting, and incentives to cut corners. This brings into focus problems of reflective stability, value faithfulness, and pressure testing. More generally, the literature may still be under-theorizing the gap between agents that merely *look* aligned and agents whose behavior remains legible under ambiguity, pressure, and strategic incentives.

## Direction 2: High-Stakes Domain AI in Education and Health

This direction is concerned with what trustworthy deployment actually demands in domains where failure is mediated through human institutions rather than isolated task scores. Education and health are especially revealing here because both domains impose constraints that current benchmark cultures tend to under-represent. In each case, a model may look strong in controlled evaluation while remaining weak along the dimensions that govern real use: calibration, fairness, temporal stability, evidence sensitivity, and behavior under uncertainty.

In education, the central gap is that **teaching quality** remains under-specified relative to answer quality. A system can be correct and still be pedagogically poor. It can scaffold at the wrong level, over-help, mis-handle uncertainty, or behave unsafely in child-facing interaction while still performing well on ordinary QA-style benchmarks. This problem appears sharper in multilingual, code-mixed, and developmentally heterogeneous settings, where current evaluations may systematically miss the instructional distortions that matter most in practice.

In health AI, the central gap is that medical usefulness cannot be reduced to isolated accuracy. Clinical readiness depends on uncertainty, provenance, workflow fit, temporal reasoning, and harm-sensitive failure. A system may answer well on exam-style tasks and still remain poorly calibrated for use in settings where abstention, escalation, evidence conflict, or longitudinal coherence matter more than a single prediction. The literature appears to be moving in this direction, but many of the relevant factors are still treated separately when they may be constitutive parts of one larger readiness problem.

Taken together, these domains suggest that benchmark success can produce a misleading picture of readiness whenever evaluation remains blind to the institutional context in which the system will actually operate. That is why questions of child-facing safety, multilingual fairness, abstention under clinical uncertainty, workflow-level robustness, provenance, and temporally extended reasoning all matter more than they may initially appear to. They are not secondary details. They are often the point at which high-stakes deployment becomes either defensible or dangerous.

## Direction 3: Verifiable Systems and Constrained Generation

This direction is motivated less by high-stakes domain deployment in the usual sense and more by the prospect of **constraint-driven program synthesis** for systems where the gains from correct automation could be substantial. The underlying intuition is straightforward: many systems tasks are governed by hard invariants, narrow execution envelopes, and explicit safety constraints. In such settings, the question is not whether a model can produce code that looks reasonable. It is whether it can move from an intent to an artifact that obeys those invariants while remaining operationally useful. The most concrete instance of this in my current portfolio is eBPF synthesis, where a generated artifact must survive compilation, verifier checks, semantic demands, and runtime conditions rather than merely look plausible.

A first open problem here concerns the gap between **intent** and **constraint-satisfying execution**. In the eBPF case, the problem is unusually stark. A user may know what they want at the level of traffic control, tracing, or enforcement, but the Linux kernel requires something much stricter: the right hook, the right helper surface, the right state assumptions, and code that passes the Linux Kernel Verifier without destabilizing the system. This makes eBPF a revealing case for invariant-driven synthesis. It forces the transition from informal objective to constrained executable artifact into the foreground.

A second open problem is evaluative rather than generative. Current code-generation benchmarks remain too close to syntax, unit tests, or surface correctness, whereas deployable constrained synthesis is governed by layered validation criteria. In the eBPF case, these include compilation, verifier acceptance, semantic correctness, and runtime behavior all at once. This suggests that the relevant evaluation target is still missing. Under these conditions, progress may be systematically overstated when verifier-safe but semantically wrong outputs are treated as near-misses rather than structural failures.

A third open problem concerns **specification elicitation**. Natural-language requests like "monitor database stalls" or "block suspicious outbound traffic" are not yet usable programming specifications. They must be mapped onto the appropriate hook choice, state model, helper surface, and safety constraints. This intermediate layer may be more central than existing accounts suggest, because it reflects a broader mechanism that recurs across AI systems: the hardest step is often not final generation, but the conversion of vague intent into a structured representation that

downstream procedures can reliably execute.

A fourth open problem concerns learning under expensive evaluators. When useful feedback requires repeated compile-and-verify loops, the cost of evaluation itself becomes part of the scientific problem. This matters not only for training efficiency, but for the deeper question of how invariant-driven synthesis should be studied when correctness is layered, operational, and costly to verify. In that sense, the eBPF setting is valuable precisely because it makes the verification problem concrete. It becomes difficult to hide behind plausibility when a synthesized program must pass the verifier and still avoid crashing the system at runtime.

## Cross-Cutting Open Problems

Across all three directions, the same questions recur with unusual persistence. How do we elicit precise operational specifications from vague human goals? How do we evaluate systems when the easiest metrics are not the ones that matter? How do we combine automated signals with human judgment without creating new opportunities for gaming? How do we stress-test models under long-horizon, multi-turn, or safety-sensitive pressure? And how do we decide when a system is ready for consequential use rather than merely good at looking capable?

If there is a single meta-project behind this agenda, it is the study of **AI systems for high-stakes, hard-to-check settings**, where the main scientific bottleneck is not only capability, but also **specification**, **verification**, and trustworthy deployment. Existing work identifies many relevant factors, but often treats them in fragmented form. My working hypothesis is that these factors are better understood as parts of one larger reliability problem, and that progress on them will transfer across domains, especially where the tolerance for hidden failure is low.

## Near-Term Open Problems

In the near term, the problems that appear most compelling are those that expose hidden reliability gaps with unusual clarity. In the alignment direction, that includes the divergence between apparent completion and real completion, the fragility of reviewer judgment under persuasive narratives, and the stability of learned constitutions under pressure. These are important not only because they diagnose current models, but because they identify mechanisms of failure that are likely to persist as models become more capable.

Within domain-facing deployment, the most important open problems concern the under-specified nature of pedagogy and clinical readiness. In education, this means whether teaching quality can be evaluated in a principled way, whether multilingual and child-facing settings reveal systematic failures that current benchmarks obscure, and whether safety remains stable over longer instructional interactions. In health, it means whether robustness under perturbation, abstention under uncertainty, workflow-level reasoning, evidence grounding, and the distribution of failure across populations are being measured with sufficient seriousness.

Within systems, the most important open problems concern invariant-driven synthesis, the specification gap between natural-language intent and constrained execution surfaces, and the cost structure of trustworthy verification itself. eBPF is useful here because it makes these issues unusually sharp. It is difficult to mistake plausible output for deployable output when verifier, runtime, and semantic constraints all matter at once.

## Closing Note

I see these topics as distinct research directions that are related, but not for the same reason. The first two are directly about **safe, trustworthy, and verifiable AI in consequential settings**. The third is about building systems for invariant- and constraint-driven program synthesis where correct automation could have large practical value, and where verification constraints are unusually explicit. What links them is not a single application story, but a recurring encounter with specification gaps, hidden failure, and the difference between plausible output and dependable behavior. That, to my mind, is where many of the most important open problems now reside.