When Not to Trust the Oracle: Detecting Unsafe Advice in LLM-guided Reinforcement Learning

Abstract

Large language models (LLMs) have recently been explored as high-level planners for reinforcement learning (RL) agents in complex environments. This approach encounters issues when we take into consideration the fact that LLM-generated advice can be unreliable, particularly under corrupted or ambiguous conditions and state descriptions. This raises the question of safety concerns for the decision-making and other processes powered by the LLM generated advice. We aim to address this gap by proposing a lightweight safety framework that detects and tries to mitigate unsafe LLM guidance.

We adopt the NetHack Learning Environment (NLE) as our testbed due to its rich combinatorial state space, partial observability, and long-horizon decision dependencies, which make it a challenging benchmark for evaluating robustness of LLM-guided agents.

Our approach integrates:

I. Rule based causal validators that capture critical survival dependencies (e.g. health, nutrition, enemy proximity),
II. Adversarial corruption tests that expose LLM failure models by perturbing the state description and the stakeholders (for example, changing the name of a character or object to confuse the context),
III. A fallback ensemble policy that defers to baseline RL agent when the unsafe advice is detected.

This work contributes a practical methodology for *trust calibration* in LLM-guided RL and opens new directions for adversarial robustness in language-driven agents.

Most research on LLM-guided reinforcement learning uses the LLM as a planner or “common sense” advisor, but leaves critical gaps. Current agents operate as black boxes: they map states to actions without understanding *why* strategies work, which limits generalization.

They also tend to *blindly trust* the LLM’s advice, even when the state description is corrupted or incomplete, leading to brittle performance. In addition, existing systems

provide little interpretability—offering no clear way to debug or understand decisions—and they are rarely stress-tested under adversarial or misleading inputs. Together, these gaps raise serious concerns about safety, reliability, and trustworthiness in real-world applications.

Project Idea:

1. Our project tackles these challenges by introducing a lightweight safety and validation framework for LLM-RL integration.
2. Instead of blindly executing suggestions, the agent checks LLM advice against simple causal or domain-knowledge–based rules to catch unsafe recommendations.
3. We also evaluate robustness by deliberately corrupting state descriptions and observing how the system responds, creating one of the first principled tests of adversarial resilience in this setting.
4. Finally, by focusing on interpretable safety checks rather than opaque policies, our approach makes it possible to understand *when* and *why* the LLM fails—offering a practical step toward building more trustworthy LLM-guided agents.