Appen

REINFORCEMENT LEARNING ENVIRONMENTS

APPEN'S METHODOLOGY & FINANCE DOMAIN DEEP DIVE

Table of contents

Appen

Executive Summary

AI agents today are executing multi-step tasks, navigating complex workflows, and making decisions in dynamic real-world environments. Reinforcement Learning (RL) environments have emerged as the leading approach to train AI agents, where agents learn by interacting with an environment and refining their behaviour based on feedback. However, the quality of that training depends entirely on the environment itself. Poorly designed RL environments produce brittle, unpredictable agents, while well-designed ones produce agents that are capable and genuinely useful in the real world.

The importance of well-constructed RL environments and frameworks is reinforced by academic research which found that a flexible RL training framework adaptable across diverse task scenarios and interactive environments lead to substantial improvements in agent performance (source: Cheng et al, 2025). Additionally, this is supported by industry commentary that the bottleneck in AI progress is no longer data, it's building RL environments that are rich, realistic, and truly useful (source: Liu et al, 2025).

This whitepaper presents Appen's rigorous, proven methodology for designing RL environments that produce high-fidelity reward signals, driving meaningful improvements in agent performance. A deep dive into the Finance domain illustrates the significant gap between current model capabilities and the demands of real-world professional workflows. When tested against pre-built, off-the-shelf Finance tasks, a SOTA model failed to pass ~88% of tasks even once across four attempts. These gaps only surface when agents are tested against environments that are reflective of real-world complexity; with organizations that invest in such environments being able to train more capable and resilient agents.

Appen's Methodology

Appen combines deep domain expertise with a rigorous, proven methodology for environment design, enabling our customers to construct high-performance RL environments at scale. Our methodology is built around two critical components that form the core building blocks of effective RL environments: Tasks and Verifiers.

Tasks

Appen has an extensive library of pre-built, off-the-shelf task sets designed for model builders who operate within their own sandboxes, tooling, and training harnesses. For organizations with specialized requirements, we also build custom task datasets tailored to specific domains, complexity levels, professional roles, and workflows.

Verifiers

Verifiers determine how model outputs are evaluated and scored, directly shaping the reward signal that drives learning. Two complementary yet mutually exclusive verifier types are supported to suit the specific requirements of a given training objective:

A) Programmatic verifiers
Automated, rule-based evaluation functions that assess agent outputs against deterministic criteria, well suited for tasks with objectively correct answers and structured outputs.

B) Rubric-based verifiers
Rubric-based verifiers are designed to deliver high-fidelity reward signals that translate directly into meaningful downstream model improvement. Each rubric is structured across multiple evaluative dimensions, with the rubric also supporting negative reward signals enabling model builders to explicitly penalize undesirable behaviours.

Appen's Methodology

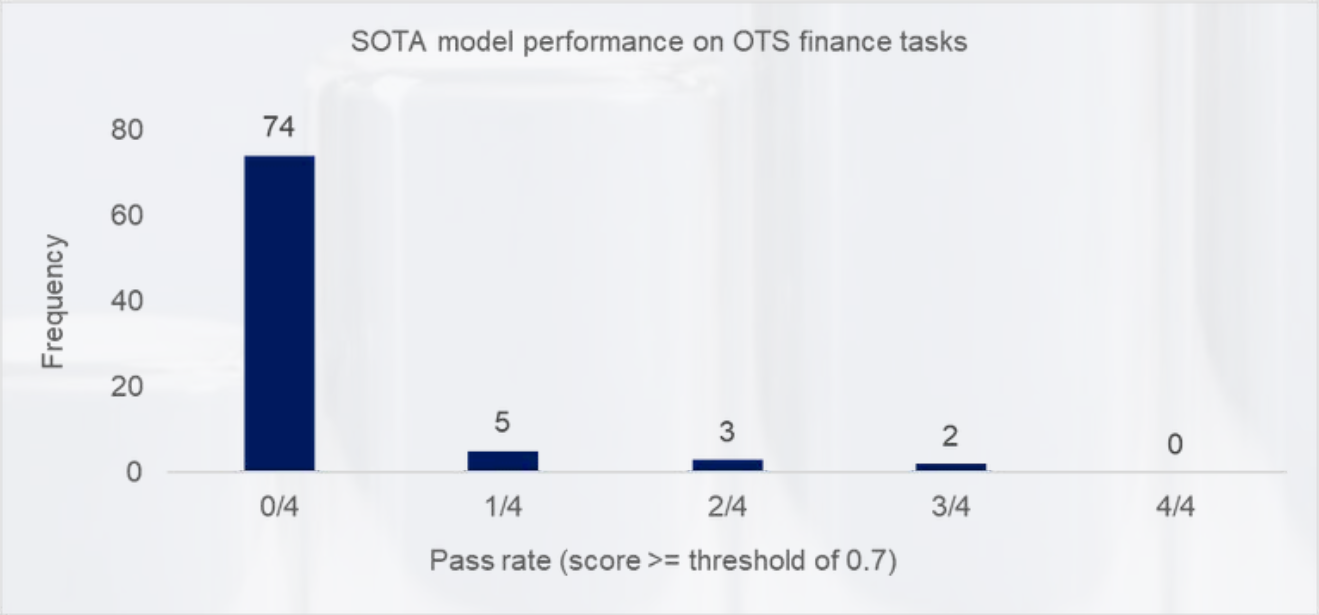
Verifiers (cont.)

Quality processes and checks are conducted on rubric-based verifiers to ensure they provide high-fidelity signals:

- Rubric refinement - First, each rubric dimension is tested for atomicity to ensure it evaluates exactly one specific aspect of a model's output. Any dimension that evaluates multiple criteria is decomposed into finer-grained components. Next, either a SOTA model or human evaluator attempts to exploit the rubric by generating or modifying trajectories that fail to accomplish the task goal yet still receive a perfect score. Any gaps exposed through this adversarial testing are incorporated as new rubric dimensions, and the entire cycle repeats until every rubric dimension is atomic and no further exploitable gaps can be found. This closed-loop process ensures rubric-based verifiers are both precisely scoped and comprehensive, leaving minimal room for reward hacking or undetected failure modes
- Scoring consistency checks – Each verifier is executed against the same model trajectory multiple times to confirm that verifier scores remain stable across all runs
- Coverage mapping – All failure modes and competencies that the rubric is intended to evaluate are listed out, ensuring that they all have a corresponding rubric dimension

Finance Domain Deep Dive

Current models find GDPval style tasks in domains such as finance and management consulting to be particularly challenging. This is illustrated through the below frequency histogram which shows the performance of a SOTA model when tested against an Off-The-Shelf (OTS) Finance task suite. The SOTA model was unable to pass at least once in 4 attempts on ~88% of the Finance tasks (i.e. 74 out of 84 finance tasks). In addition, the OTS model was able to pass once in 4 attempts on only ~6% of the Finance tasks (i.e. 5 out of 84 finance tasks).



Appen

Finance Domain Deep Dive

An example of a task that the SOTA model failed to pass from the OTS Finance task suite is listed in the below table to demonstrate the complexity of the pre-built tasks, along with identified failure modes to detail out exactly where the SOTA model failed. Here, the SOTA model scored ~65% based on automated evaluations so failed to pass because the passing threshold for this task was 70%.

The automated evaluations cover both programmatic verifiers (e.g., numeric accuracy against a golden answer) and rubric-based verifiers (e.g., structure and coherence of the output, coverage of topics). These evaluations revealed particularly low scores on categories such as structure which only scored ~7%. While these evaluations can reliably flag failures, these automated checks do not reveal the reason behind it. This is where human expert review of trajectories adds value by uncovering the rationale behind both failures and successes, providing a higher-fidelity signal on where to improve the model.

Task objective	• Construct a single Excel workbook with 10 required sheets for APPLE (AAPL) that isolates genuine foreign exchange impact from accounting distortions • The workbook should enable an analyst to quickly determine whether an FX-driven EPS miss reflects a real economic effect or merely an accounting artifact, and whether such patterns have historically tended to reverse
Identified failure modes	• Incomplete output – SOTA model created a workbook with only 4 sheets that were partially populated, whilst the task requirements specified a workbook with 10 fully populated sheets

Finance Domain Deep Dive

Identified failure modes	• Incoherent narrative – SOTA model did not tell a coherent and synthesized story on the implications of a FX-driven EPS miss, as it did not have an "Executive Dashboard" summary sheet that synthesized metrics from all downstream detailed sheets • Lack of source documentation – SOTA model failed to provide any specific SEC filing source citations and disclosed assumptions in its workbook, which was a critical task requirement • Incorrect financial analysis – SOTA model performed incorrect financial analysis when it came to computing FX impacts and developing the hedge separation framework

Conclusion

The findings presented in this paper make one thing clear: the gap between what today's frontier models can do and what real-world tasks demand is both significant and measurable. Critically, these gaps only become visible when agents are tested against well-constructed RL environments that faithfully represent the complexity of real-world workflows. Without that fidelity in environment design, teams risk training agents that perform well in controlled settings but fail unpredictably when deployed.

Organizations that invest in rigorous, domain-grounded environments whether through Appen's off-the-shelf offerings or custom-built solutions tailored to their specific workflows; will train agents that are more capable, more resilient, and more aligned with the tasks they are designed to perform.

At Appen, we are committed to partnering with teams at the frontier of agent development, providing the environment design expertise and high-quality training infrastructure needed to close the gap between where AI agents are today and where they need to be.

Talk to an Expert

Explore how to improve your agents with our RL environments

appen.com | sales@appen.com

Explore what we do

Appen