PUSHING THE LIMITS OF LABORATORY 3D PERCEPTION AND LONG-HORIZON PLANNING VIA PROTOCOL-ALIGNED ACTION PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

AI Scientist systems are moving toward physical laboratory operation, which requires asset understanding, protocol alignment, and long-horizon state tracking beyond general scientific knowledge. We introduce **LabHorizon**, a training and evaluation framework for **Protocol-Aligned Action Prediction**. Level 1 predicts the next action from multi-view laboratory assets and protocol context; Level 2 generates parameterized action sequences from protocol windows and action pools. The release contains **6,400** samples: 3,000 training and 200 test samples per level, spanning 108 Level-1 assets and 2,342 unique wet lab protocols. Evaluations across major model families reveal persistent failures in asset-conditioned action selection, numerical parameters, action order, and variable dependencies. We train a Qwen3.6-35B-A3B domain model and build an Actor-Simulator-Selector agent. Training improves Level 1 accuracy from 47.5% to 63.5% and Level 2 final score from 25.34% to 41.00%; the agent further improves them to 66.5% and 45.32%. LabHorizon thus provides diagnostic evaluation and learnable supervision for laboratory action prediction.



Figure 1: **LabHorizon overview.** Multi-view laboratory assets and real wet lab protocols support next-action prediction, action-sequence planning, evaluation, and a training–agent learning loop.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

AI Scientist systems are moving from digital reasoning to physical wet lab operation (Boiko et al., 2023; M. Bran et al., 2024; Lu et al., 2024a; Jansen et al., 2024; Cong et al., 2025). Beyond literature and hypotheses, they must understand laboratory assets, track sample states, and preserve protocol-specific temperatures, durations, volumes, speeds, and intermediate products. This connects scientific agents to embodied intelligence: perception must support actions grounded in laboratory workflows and protocol-local experimental states.

Existing knowledge benchmarks (Hendrycks et al., 2020; Rein et al., 2023; Du et al., 2026), scientific multimodal datasets (Lu et al., 2022; Yue et al., 2024; Zhao et al., 2025), and planning resources (Valmeekam et al., 2023; Xu et al., 2025; Duan et al., 2026) address parts of this problem, but rarely connect multi-view laboratory assets, real protocol constraints, long-horizon action prediction, and matched training/evaluation supervision.

We introduce **LabHorizon** and formulate both levels as **Protocol-Aligned Action Prediction** (figure 1). **Level 1** predicts a next action from three asset views, historical actions, and near-miss candidates. The views provide complementary global-layout and local-affordance evidence inspired by vision-language-action (VLA) systems (Wang et al., 2026; Li et al., 2026a; Gao et al., 2026). **Level 2** generates a parameterized action sequence from a real protocol window and an action pool; Python Abstract Syntax Tree (AST) parsing (Crew et al., 1997) evaluates action order, named action parameters, and dependencies between intermediate experimental products.

The framework addresses four challenges: **Laboratory 3D Perception**, **Real-World Protocol Alignment**, **Long-Horizon Planning**, and an **Optimizable Learning Loop**. Strong models confuse nearby stages, replace protocol values with defaults, omit or duplicate actions, and break variable dependencies. We train a Qwen3.6-35B-A3B domain model and introduce an **Actor-Simulator-Selector** agent that checks symbolic future-state consistency before selecting an action prediction.

Contributions. (1) A 6,400-sample framework spanning 108 Level-1 assets and 2,342 unique protocols, with 3,000 training and 200 test samples per level. (2) A unified task connecting multi-view asset-to-action prediction and action-pool planning. (3) Evaluation and diagnostics across major open and closed model families. (4) A training–agent loop: training improves Level 1 accuracy from 47.5% to 63.5% and L2 final score from 25.34% to 41.00%; adding Actor-Simulator-Selector further improves the same two metrics to 66.5% and 45.32%, respectively.

2 RELATED WORK

2.1 SCIENTIFIC MULTIMODAL DATA FOR AI SCIENTIST SYSTEMS

ScienceQA (Lu et al., 2022), MathVista (Lu et al., 2024b), MMMU/MMMU-Pro (Yue et al., 2024; 2025), CMMMU/CMMU (Zhang et al., 2024; He et al., 2024b), and OlympiadBench (He et al., 2024a) evaluate scientific, mathematical, or expert multimodal reasoning. SPIQA, CharXiv, and MSEarth study scientific papers, charts, and Earth-science evidence (Pramanick et al., 2024; Wang et al., 2024; Zhao et al., 2025). MME-SCI, MMSciBench, CSVQA, mmJEE-Eval, and RBench/RBench-V broaden scientific domains and reasoning settings (Ruan et al., 2026; Ye et al., 2025; Jian et al., 2025; Mukherjee & Ghosh, 2025; Guo et al., 2025); MMRB, Kaleidoscope, and EncQA add multi-image, multilingual, and visual-encoding tasks (Cheng et al., 2025; Salazar et al., 2025; Mukherjee et al., 2025). Unlike interpreting curated scientific representations, LabHorizon asks what action a pictured laboratory asset supports in a local wet lab state.

2.2 LONG-HORIZON EXPERIMENTAL PLANNING

Scientific agents support discovery, literature, tool use, and experimental design (Boiko et al., 2023; M. Bran et al., 2024; Lu et al., 2024a; Yamada et al., 2025; Ghafarollahi & Buehler, 2025; Roohani et al., 2025; Lála et al., 2023; Jansen et al., 2024); evaluations also cover machine-learning experimentation, data-driven science, research, and replication (Huang et al., 2023; Gu et al., 2024; Chen et al., 2025; Chan et al., 2025; Wijk et al., 2024; Starace et al., 2025). Formal planning and reasoning studies (Valmeekam et al., 2023; Liu et al., 2023; Kambhampati et al., 2024; Kambhampati,

2024; Valmeekam et al., 2024; Lin et al., 2024), tool/web/computer environments (Xie et al., 2024a; Zhou et al., 2024; Liu et al., 2024; Ma et al., 2024; Xie et al., 2024b), and structured planning benchmarks (Li et al., 2025; Zuo et al., 2025; Anokhin et al., 2025; Wang et al., 2025; Kokel et al., 2025; Burigana & Fabiano, 2026) test capabilities distinct from the dense temperatures, durations, volumes, speeds, and sample-state dependencies of wet lab protocols.

Closer resources include biological experiment evaluation (OpenAI, 2025), the systems-biology dry lab SciGym (Duan et al., 2026), SGI-WetExperiment action pools (Xu et al., 2025), and protocol comprehension, troubleshooting, and planning in LAB-Bench/LABBench2, BioLP-bench, and BioProBench (Laurent et al., 2024; 2026; Ivanov, 2024; Liu et al., 2025). ProtocolBench instead studies multi-agent communication protocols (Du et al., 2025). Embodied VLA work connects visual observations to robot actions (Wang et al., 2026; Li et al., 2026a; Gao et al., 2026). LabHorizon targets the semantic protocol-aligned action layer, combining multi-view assets, local numerical constraints, structured plans, and matched train/test splits. Appendix A, table 3, compares these dimensions using public task definitions: functional laboratory-asset perception, wet-protocol-local alignment, multi-step wet experiment planning, and support for training, diagnosis, and agent iteration. This framework does not claim to subsume the full range of existing scientific benchmarks.

3 METHOD

3.1 TASK FORMULATION

LabHorizon connects asset perception and experimental planning through two levels of Protocol-Aligned Action Prediction. Level 1 predicts a next action:

$$\mathcal{T}_1 : x_1 = (P, \mathcal{I}, h, \mathcal{C}) \to y_1 = (r, n), \quad \mathcal{I} = \{I_1, I_2, I_3\}, \quad \mathcal{C} = \{c_1, \ldots, c_{10}\}, \quad n \in \mathcal{C}. \tag{1}$$

P is the protocol condition, $\mathcal{I}$ contains three views of one asset, h records historical actions and the current sample state, $\mathcal{C}$ contains ten candidates, and (r, n) gives reasoning and the selected action. Complementary views expose global layout and local affordances, a static analogue of the complementary external-camera and wrist/egocentric observations used by VLA systems.

Level 2 generates a complete action-pool plan:

$$\mathcal{T}_2 : x_2 = (P, c, g, R, U, \mathcal{A}) \to y_2 = \mathbf{s} = (s_1, \ldots, s_T), \quad s_t = (a_t, \mathbf{p}_t, z_t). \tag{2}$$

Here c is context, g the goal, R constraints, U raw inputs, and $\mathcal{A}$ the action pool. Step s_t contains action a_t, parameters $\mathbf{p}_t$, and output variable z_t, which later actions can consume. The four-stage pipeline (figure 2) organizes assets/protocols/action libraries, constructs level-specific targets, applies difficulty and quality controls, and releases matched train/test splits.

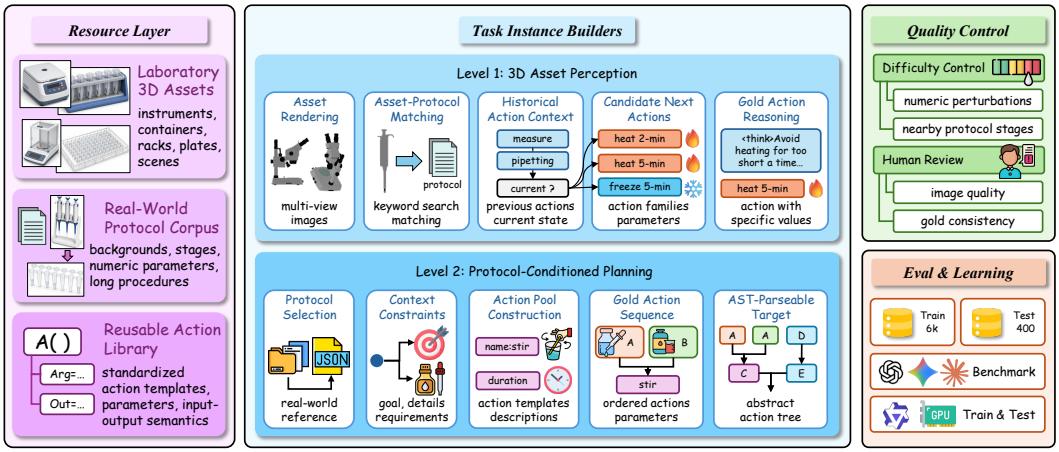


Figure 2: **LabHorizon construction pipeline.** Assets, protocols, and action libraries yield Level 1 asset-to-action samples and Level 2 action-pool plans. Difficulty controls, automatic schema/leakage/dependency checks, and human review precede train/test release and model training.

3.2 Level 1 Construction: Laboratory 3D Asset Perception

Each sample contains three asset views, historical context, ten candidates, reference reasoning, and a gold next action. Instruments, consumables, containers, racks, plates, and scene objects draw on resources such as AutoBio and LabUtopia (Lan et al., 2025; Li et al., 2026b). Rendered or selected images must be upright, clear, and informative. Asset affordances are aligned with local protocol stages: centrifuges with centrifugation, thermal cyclers with temperature programs, magnetic racks with bead capture, and ice buckets with cold holding. History records completed actions and sample state while avoiding direct leakage of the asset name, family label, or gold next action.

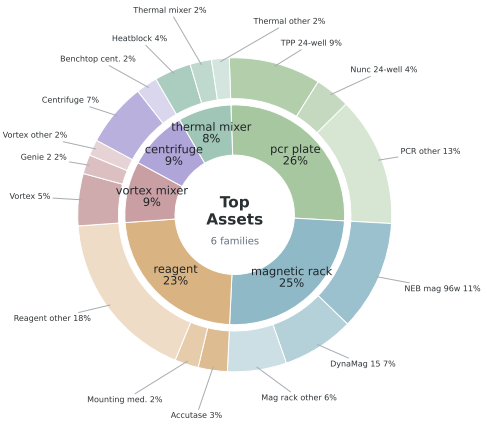


Figure 3: **Level 1 asset families.** The inner ring shows the six most frequent families; the outer ring shows representative assets and labeled residual groups, not the full 108-asset inventory.

Nine candidates form three action families, each with three parameter variants sharing an action name and schema; the tenth is `none_of_the_above`, which is sometimes gold. Distractors use nearby protocol stages and subtle numerical changes. Gold families contain same-action competitors; same-family distractors usually change one key parameter. Neither naming nor keyword matching suffices.

Validators check English-only text, candidate schemas, gold/reasoning consistency, image paths, asset-name leakage, off-context distractors, empty operands, and unsupported affordances. Manual audit checks image quality, semantic correctness, and whether the pictured asset supports the action. The six displayed families in figure 3 are PCR plates (26%), magnetic racks (25%), reagents (23%), vortex mixers (9%), centrifuges (9%), and thermal mixers (8%); the full release records every asset, every selected view, and every multi-view image reference in the Level 1 dataset.

3.3 Level 2 Construction: Protocol-Aligned Long-Horizon Planning

We select local windows from real protocols and specify context, goals, constraints, raw inputs, and focused action pools. Templates provide descriptions and input/output semantics; identical actions reuse the same template. Gold sequences assign action outputs to variables consumed downstream, preserving order, parameters, intermediate products, and branch topology. Windows contain 8–13 standardized actions with repeated families, nearby-stage ambiguity, and numerical constraints. The working corpus has 3,749 protocols averaging about 77.5 raw steps, including 890 with at least 100 steps; the standardized action sequences are shorter than these source protocols.

Strong models draft candidate samples. Automatic validators check English-only text, action existence, parameter schemas, image paths, leakage, missing actions, invalid parameters, undefined variables, and dependency chains; human review follows. Low-quality samples are repaired or regenerated. Multi-model probing additionally checks whether models exhibit redundant actions, action-order errors, numerical parameter drift, and broken variable-dependency chains.

3.4 Data Statistics

Each level has 3,000 training and 200 test samples (table 4, Appendix A). Both splits use the same construction and quality controls. Level 1 shares 108 assets across splits but uses different protocol contexts and prediction instances. Level 2 chains average 12.61 training and 12.57 test actions (maximum 13). The release spans 2,342 unique protocols.

3.5 Structured Evaluation Metrics

Level 1 Next Action Accuracy is the fraction whose extracted final action matches the gold candidate; natural-language reasoning is allowed. Level 2 extracts the final structured sequence and uses Python AST parsing (Xu et al., 2025; Crew et al., 1997) to identify calls, named parameters, assignments, and variable references. It detects nonexistent actions, illegal keys, undefined variables, literal strings replacing generated variables, and broken dependencies while recognizing equivalent action-dependency topology even when the variable identifiers differ.

For gold $\mathbf{s} = (s_1, \ldots, s_T)$ and prediction $\hat{\mathbf{s}} = (\hat{s}_1, \ldots, \hat{s}_{\hat{T}})$, parsed steps have the form $(a_t, \mathbf{p}_t, z_t)$. **L2 Action Sequence Similarity** measures action-order recovery:

$$\mathrm{ASS}(\mathbf{s}, \hat{\mathbf{s}}) = \frac{\sum_{t=1}^{\min(T,\hat{T})} \mathbb{I}[a_t = \hat{a}_t]}{\max(T, \hat{T})}. \tag{3}$$

The denominator penalizes missing and redundant actions while allowing partial credit for recovered fragments. **L2 Parameter Accuracy** uses a monotonic action-name alignment $\mathcal{M}$, a map ϕ from aligned predicted outputs to gold variables, and the union K_{ij} of parameter keys for each aligned pair, including keys present only in the prediction or only in the gold action:

$$\mathrm{PA}(\mathbf{s}, \hat{\mathbf{s}}) = \frac{\sum_{(i,j)\in\mathcal{M}} \sum_{k\in K_{ij}} \mathbb{I}[\mathrm{match}(p_{i,k}, \hat{p}_{j,k}; \phi)]}{\sum_{(i,j)\in\mathcal{M}} |K_{ij}| + \sum_{i\notin\mathcal{M}_g} |\mathrm{keys}(\mathbf{p}_i)| + \sum_{j\notin\mathcal{M}_p} |\mathrm{keys}(\hat{\mathbf{p}}_j)|}. \tag{4}$$

Matching checks key presence, types, literals, numerical values under a unified tolerance, raw-input references, and generated-variable dependencies under ϕ. Unaligned steps contribute only to the denominator. The final score is the arithmetic mean of these two Level 2 metrics:

$$\mathrm{L2FinalScore} = \frac{\mathrm{ASS} + \mathrm{PA}}{2}. \tag{5}$$

3.6 Actor-Simulator-Selector Agent

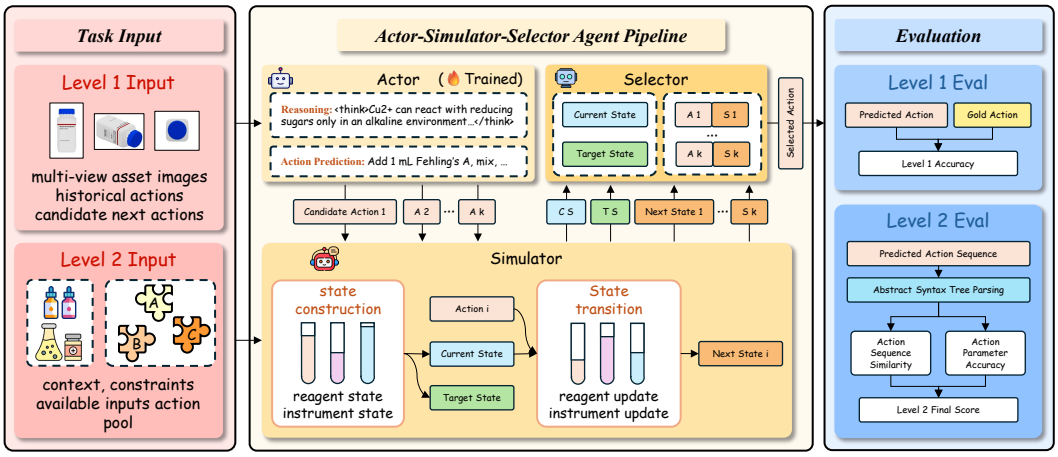


Figure 4: **Actor-Simulator-Selector.** The Actor proposes actions from level-specific inputs; the Simulator predicts symbolic state transitions; the Selector compares candidates with the target state. Outputs are evaluated by Level 1 accuracy or Level 2 AST-based metrics.

The agent (figure 4) combines simulation-enabled planning (Liu et al., 2026) with world-model (Tu et al., 2026; Team et al., 2026; Yan et al., 2026; Hou et al., 2026) and world-action-model (Yuan et al., 2026) ideas. Here future state is symbolic: produced variables, consumed samples, reagent/instrument states, stage order, and parameter constraints. The **Actor** reads level-specific inputs and samples K candidate next actions or complete sequences with reasoning. The **Simulator** constructs current and target states and predicts each candidate's reagent/instrument updates. The **Selector** compares candidate–state pairs (A_i, S_i) with the target and selects the most consistent prediction.

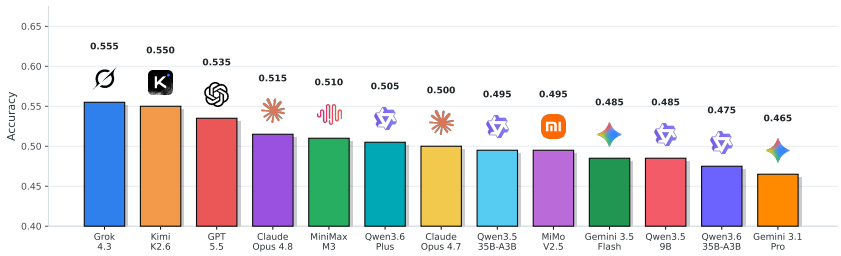


Figure 5: **Level 1 direct prompting.** Next Action Accuracy on 200 test samples, sorted highest to lowest; the value above each bar reports exact accuracy over all 200 samples.

Candidates may be down-ranked for out-of-pool actions, positional arguments, undefined variables, broken sample-state chains, physically impossible stages, or numerical drift. This complements training: learned action interfaces and protocol patterns support generation, while symbolic future-state checks improve the consistency of action predictions at inference time.

3.7 Domain Model Training

We fine-tune Qwen3.6-35B-A3B (Qwen Team, 2026) with LoRA (Hu et al., 2022) on 3,000 three-image Level 1 and 3,000 text-only Level 2 samples. Qwen chat data are converted to LLaMA-Factory (Zheng et al., 2024) ShareGPT-VL-style format inspired by ShareGPT4V (Chen et al., 2024). Settings are rank 32, alpha 64, dropout 0.10, cutoff 4096, image max pixels 501760, learning rate 1.0×10^{-4}, cosine schedule, warmup 0.10, 10 epochs, 2500 max steps, bf16, and gradient checkpointing. Runtime is 10014.77 s; final train/eval losses are 0.2691/0.4426. The run tests learnable operation signals, not model saturation; full settings and logs are in Appendix B, table 5.

4 Experiments

4.1 Experimental Setup

We use the formal test splits: 200 Level 1 multi-view samples and 200 Level 2 plans. Natural-language responses are allowed without forced JSON or a conservative output-length cap, avoiding reasoning truncation. Evaluators extract the final candidate or AST-parsed sequence. The suite includes GPT-5.5, Claude Opus 4.7/4.8, Gemini 3.1 Pro/3.5 Flash, Grok 4.3, Qwen, GLM 5.1, Kimi K2.6, MiMo V2.5, MiniMax M3, and DeepSeek V4 Pro (table 1). Some text-only models are tested only on Level 2. Direct, trained, and trained+agent settings share the same test splits.

4.2 Level 1: Laboratory 3D Perception

Accuracies concentrate around 0.46–0.56 (figure 5): Grok 4.3 scores 0.555, GPT-5.5 0.535, Claude Opus 4.8 0.515, MiniMax M3 0.510, Claude Opus 4.7 0.500, and Gemini 3.1 Pro 0.465. Models often recognize assets but cannot bind visual affordances, local protocol stages, and exact candidate parameters. Typical errors replace a protocol-given 9 C with 4 C or a 3 min magnetic-rack duration with 4 min. Level 1 therefore tests protocol-conditioned functional understanding across visual, contextual, and candidate-action evidence rather than object naming alone.

4.3 Level 2: Long-Horizon Experimental Planning

The strongest direct models reach final scores near 0.32 (table 1): Gemini 3.1 Pro 0.3263, Grok 4.3 0.3244, and Kimi K2.6 0.3150. MiniMax M3 (0.2954) and Claude Opus 4.8 (0.2911) exceed Opus 4.7 but trail the leaders; Qwen3.5 9B scores 0.1315. Low sequence and parameter scores reflect more than formatting: models recover local digestion, centrifugation, collection, or cycling fragments yet omit wash/quench/settle/resuspend/transfer steps, duplicate stages, substitute default parameters, and break intermediate-product chains. The bottleneck is global consistency and sample-state tracking over the whole experimental plan, not the absence of local laboratory action knowledge.

Table 1: **Level 2 direct-prompting results on the 200-sample test split.** Higher is better. Darker cells indicate higher values within each metric; bold marks best and underline marks second-best.

Model	L2 Action Sequence Similarity ↑	L2 Parameter Accuracy ↑	L2 Final Score ↑
Gemini 3.1 Pro (Google, 2026)	0.3195	0.3331	**0.3263**
Grok 4.3 (xAI, 2026)	**0.3339**	0.3148	0.3244
Kimi K2.6 (Team et al., 2025)	0.2845	**0.3456**	0.3150
Gemini 3.5 Flash (Google, 2026)	0.2686	0.3391	0.3039
Qwen3.7 Max (Yang et al., 2025)	0.2905	0.3102	0.3003
MiniMax M3 (MiniMax, 2026)	0.2812	0.3095	0.2954
Claude Opus 4.8 (Anthropic, 2026)	0.2756	0.3066	0.2911
Claude Opus 4.7 (Anthropic, 2026)	0.2619	0.2856	0.2737
Qwen3.6 35B-A3B (Qwen Team, 2026)	0.2585	0.2483	0.2534
Qwen3.6 Plus (Yang et al., 2025)	0.2264	0.2787	0.2526
MiMo V2.5 (Xiaomi MiMo Team, 2026)	0.2269	0.2713	0.2491
GLM 5.1 (Zeng et al., 2026)	0.2307	0.2519	0.2413
Qwen3.5 35B-A3B (Yang et al., 2025)	0.2385	0.2398	0.2391
GPT-5.5 (Singh et al., 2025)	0.2092	0.2459	0.2276
DeepSeek V4 Pro (DeepSeek-AI, 2026)	0.1927	0.2342	0.2135
Qwen3.5 9B (Yang et al., 2025)	0.1359	0.1271	0.1315

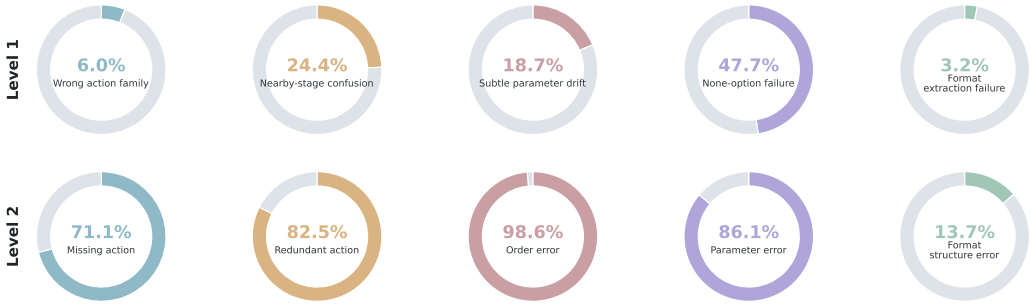


Figure 6: **Error diagnostics.** Level 1: mutually exclusive primary labels among incorrect responses (sum ≈100%). Level 2: non-exclusive tags among all planning responses (sum can exceed 100%).

4.4 Error-Type Diagnostics

Level 1. Among 2,600 responses from 13 image-capable models, 1,314 are correct. The remaining 1,286 each receive one primary label (figure 6): none-option failure 47.7%, nearby-stage confusion 24.4%, subtle parameter drift 18.7%, wrong action family 6.0%, and format/extraction failure 3.2%. None-option failures either select `none_of_the_above` despite a valid candidate or miss it when all concrete candidates are wrong. These are proportions of errors, not all responses; exact-match calibration and local-stage alignment dominate parsing failures.

Level 2. All 3,200 responses from 16 models receive non-exclusive tags: order error 98.6%, parameter error 86.1%, redundant action 82.5%, missing action 71.1%, and format/structure error 13.7%. Rates can exceed 100% because errors co-occur. Parseable plans may still lose sample identity, repeated wash/spin/incubation stages, and numerical constraints across 8–13 actions, motivating matched training and structured consistency checks.

4.5 Training and Actor-Simulator-Selector Ablation

Training provides the main gain (table 2): Level 1 rises from 47.5% to 63.5%, and L2 final score from 25.34% to 41.00%. Actor-Simulator-Selector further raises them to 66.5% and 45.32%, selecting more complete plans with stabler dependencies and fewer obvious parameter-range violations. The data mixture and LoRA configuration are unchanged from Section 3 and Appendix B. Training teaches interfaces and local protocol patterns; the agent adds inference-time consistency. Exact

Table 2: **Training and agent comparison on the same test splits.** Rows cover direct leaders, representative baselines, and the Qwen ablation. *Trained Qwen3.6-35B-A3B is Actor; Gemini 3.1 Pro is Simulator/Selector. This choice is not exhaustively ablated.

System	Level 1 Acc. ↑	L2 ASS ↑	L2 PA ↑	L2 Final Score ↑	Notes
Grok 4.3 (xAI, 2026)	0.555	0.3339	0.3148	0.3244	direct Level 1 SOTA
Gemini 3.1 Pro (Google, 2026)	0.465	0.3195	0.3331	0.3263	direct Level 2 SOTA
GPT-5.5 (Singh et al., 2025)	0.535	0.2092	0.2459	0.2276	representative direct baseline
Kimi K2.6 (Team et al., 2025)	0.550	0.2845	0.3456	0.3150	representative direct baseline
Qwen3.6-35B-A3B (Qwen Team, 2026)	0.475	0.2585	0.2483	0.2534	direct prompting
Qwen3.6-35B-A3B(trained) (Qwen Team, 2026)	0.635	0.4030	0.4170	0.4100	LabHorizon train split
Qwen3.6-35B-A3B(trained+agents*) (Qwen Team, 2026)	**0.665**	**0.4485**	**0.4580**	**0.4532**	trained model plus agent

long-horizon recovery remains difficult, and the Simulator/Selector choice is not exhaustively ablated: this is a consistency layer, not a complete solution.

4.6 Analysis

Complementary diagnostic axes. Among common models, Gemini 3.1 Pro ranks last on Level 1 (0.465) but first on Level 2 (0.3263; figure 7). Level 1 requires visual grounding and exact candidate calibration, whereas Level 2 stresses text-conditioned state tracking and dependencies. Gemini predicts `none_of_the_above` 74 times although it is gold only 40 times; just 15 of those 74 predictions are correct. It can recognize vortex mixing yet reject the protocol's 9 C candidate in favor of a 4 C prior. On Level 2 it has only 5 invalid responses out of 200 and often recovers long action skeletons despite imperfect parameters. Thus the levels should not be read as a single monotonic ranking.

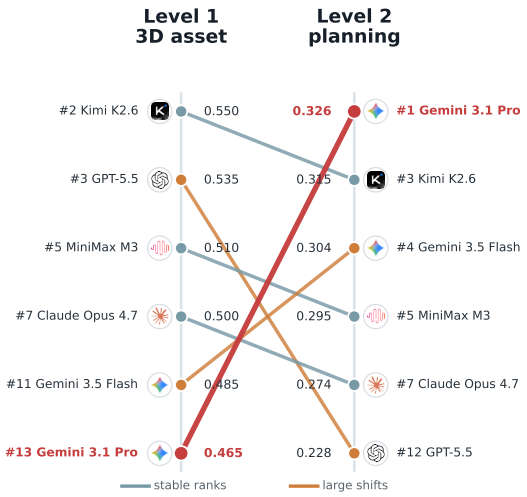


Figure 7: **Cross-level rank shifts.** Three stable and three large-shift examples; ranks use all models evaluated on both levels. Gemini 3.1 Pro reverses from last on Level 1 to first on Level 2.

State and parameter drift. Asset category recognition does not establish an asset's action role in the current protocol. Long plans must also distinguish raw sample, pellet, supernatant, cooled plate, and final product. Models may omit required steps, merge adjacent stages, or repeat the same local stage. Changing 525 x g to 500 x g, 4 C to room temperature, or 100 uL to 200 uL can alter an experiment. Replacing an intermediate variable with a literal or raw input breaks dependencies. Structured evaluation exposes errors hidden by natural-language similarity; matched training data and structured agent checks turn these repeated failures into concrete targets for protocol-aligned model improvement.

4.7 Case Study

Figure 8 shows complementary failures. In Level 1, a vortex-mixer case follows Ni-NTA resin prewashing, low-temperature incubation with eluate and 10 mM imidazole, centrifugal collection, and three T300/T150 washes. Claude Opus 4.8 correctly selects `vortex_elution` but replaces the protocol's 200 mM imidazole with 300 mM: a plausible laboratory prior overrides the protocol-specific local numeric value despite correct visual recognition of the asset.

The Level 2 Golden Gate case covers assembly thermal cycling: plate sealing, lid handling, loading/cooling, a 20-cycle 37 C / 16 C thermal-cycler program, post-run cooling, and storage. The gold action sequence preserves the intended action topology: seal the plate, open the lid, cool/load the

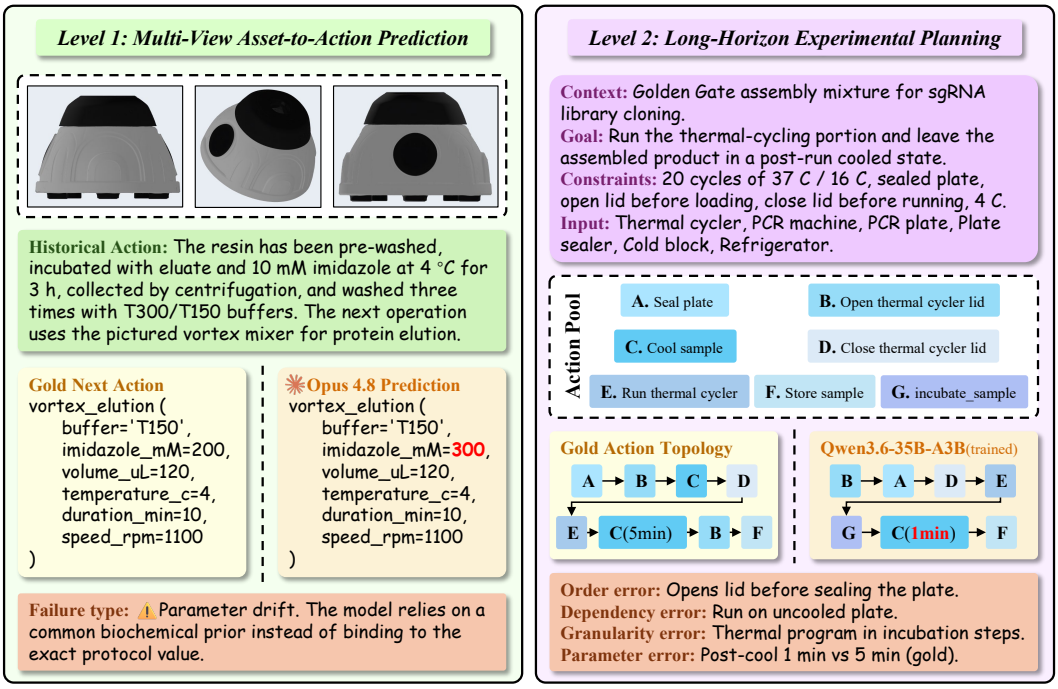


Figure 8: **Representative failures.** Left: Claude Opus 4.8 selects `vortex_elution` but changes 200 mM imidazole to 300 mM. Right: trained Qwen3.6-35B-A3B produces parseable thermal-cycler actions with incorrect order, dependencies, granularity, and cooling duration.

sealed plate, close the lid, run the predefined thermal-cycler program, cool the thermocycled product for 5 min, reopen the lid, and store the cooled product. Trained Qwen produces parseable output, but opens the lid before sealing the plate, runs the program on the wrong intermediate variable, expands the abstract thermal-cycler program into local incubation calls, and shortens post-run cooling from 5 min to 1 min. These errors lower both L2 Action Sequence Similarity and L2 Parameter Accuracy by violating stage order, action granularity, and the dependency chain. Appendix C retains complete candidates, action pools, gold sequences, reasoning, and scores.

5 Conclusion

We presented LabHorizon, a training and evaluation data framework for real laboratory operation. The central task formulation is Protocol-Aligned Action Prediction: given real protocol context, a model must predict laboratory actions, parameters, and long-horizon action sequences that are consistent with visual assets, experimental state, and available action pools. This framing connects Laboratory 3D Perception, Real-World Protocol Alignment, Long-Horizon Planning, and an Optimizable Learning Loop for laboratory action prediction and model improvement.

The current release contains 6,400 samples, including matched train/test splits for both levels. Level 1 evaluates multi-view asset-conditioned next-action prediction; Level 2 evaluates action-pool long-horizon experimental planning. Evaluations across major model families show that current models remain brittle: they confuse asset function with object naming, mix adjacent protocol stages, drift on numerical parameters, omit or duplicate steps, and break intermediate-variable dependencies.

We further showed that the data are useful for model improvement. LoRA training of Qwen3.6-35B-A3B substantially improves both Level 1 and Level 2, and the Actor-Simulator-Selector agent adds a further structured-consistency gain. These results suggest that real laboratory operation is not an automatic consequence of general scientific knowledge ability or generic multimodal understanding. Future work will expand asset coverage, include longer action windows, deepen the agent ablation, and connect protocol-aligned planning with embodied wet lab assistants and real or simulated laboratory execution systems for future AI Scientist agents.

AI USE STATEMENT

We used AI tools to assist with writing and polishing the manuscript, as well as LaTeX template conversion and formatting. Generative models were also used to draft candidate samples during data construction, followed by the automatic and human quality checks described in Section 3. The evaluated models and the Actor-Simulator-Selector agent are part of the research methodology described in Sections 3 and 4.

ETHICS STATEMENT

The ethics, social impact, intended use, and safety boundaries of LabHorizon are discussed in Appendix E.

REPRODUCIBILITY STATEMENT

Data construction, structured evaluation, and the agent are described in Section 3; experimental settings and results are described in Section 4. Training hyperparameters are provided in Appendix B, with detailed cases in Appendix C.

REFERENCES

Petr Anokhin, Roman Khalikov, Stefan Rebrikov, Viktor Volkov, Artyom Sorokin, and Vincent Bissonnette. Herobench: A benchmark for long-horizon planning and structured reasoning in virtual worlds. *arXiv preprint arXiv:2508.12782*, 2025.

Anthropic. Claude API: Models overview, 2026. Accessed 2026-06-02.

Daniil A Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023.

Alessandro Burigana and Francesco Fabiano. The epistemic planning domain definition language: Official guideline. *arXiv preprint arXiv:2601.20969*, 2026.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. In *International Conference on Learning Representations*, volume 2025, pp. 50466–50494, 2025.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, et al. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations*, volume 2025, pp. 96934–96990, 2025.

Ziming Cheng, Binrui Xu, Lisheng Gong, Zuhe Song, Tianshuo Zhou, Shiqi Zhong, Siyu Ren, Mingxiang Chen, Xiangchao Meng, Yuxin Zhang, et al. Evaluating mllms with multimodal multi-image reasoning benchmark. *arXiv preprint arXiv:2506.04280*, 2025.

Le Cong, David Smerkous, Xiaotong Wang, Di Yin, Zaixi Zhang, Ruofan Jin, Yinkai Wang, Michal Gerasimiuk, Ravi K Dinesh, Alex Smerkous, et al. Labos: The ai-xr co-scientist that sees and works with humans. *arXiv preprint arXiv:2510.14861*, 2025.

Roger F Crew et al. Astlog: A language for examining abstract syntax trees. In *DSL*, volume 97, pp. 18–18, 1997.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Hongyi Du, Jiaqi Su, Jisen Li, Lijie Ding, Yingxuan Yang, Peixuan Han, Xiangru Tang, Kunlun Zhu, and Jiaxuan You. Which llm multi-agent protocol to choose? *arXiv preprint arXiv:2510.17149*, 2025.

Xeron Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, et al. Supergpqa: Scaling llm evaluation across 285 graduate disciplines. *Advances in Neural Information Processing Systems*, 38, 2026.

Haonan Duan, Stephen Lu, Caitlin F Harrigan, Nishkrit Desai, Jiarui Lu, Michał Koziarski, Leonardo Cotta, and Chris Maddison. Measuring scientific capabilities of language models with a systems biology dry lab. *Advances in Neural Information Processing Systems*, 38, 2026.

Mingjian Gao, Wenqiao Zhang, Yuqian Yuan, Yang Dai, Binhe Yu, Zheqi Lv, Haoyu Zheng, Jiaqi Zhu, Zhiqi Ge, Zixuan Wan, et al. Visualthink-vla: Visual intermediate reasoning for effective and low-latency vision-language-action policies. *arXiv preprint arXiv:2605.30011*, 2026.

Alireza Ghafarollahi and Markus J Buehler. Sciagents: automating scientific discovery through bioinspired multi-agent intelligent graph reasoning. *Advanced Materials*, 37(22):2413523, 2025.

Google. Gemini API: Gemini models, 2026. Accessed 2026-06-02.

Ken Gu, Ruoxi Shang, Ruien Jiang, Keying Kuang, Richard-John Lin, Donghe Lyu, Yue Mao, Youran Pan, Teng Wu, Jiaqian Yu, et al. Blade: Benchmarking language model agents for data-driven science. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13936–13971, 2024.

Meng-Hao Guo, Jiajun Xu, Yi Zhang, Jiaxi Song, Haoyang Peng, Yi-Xuan Deng, Xinzhi Dong, Kiyohiro Nakayama, Zhengyang Geng, Chen Wang, et al. R-bench: Graduate-level multi-disciplinary benchmarks for llm & mllm complex reasoning evaluation. *arXiv preprint arXiv:2505.02018*, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024a.

Zheqi He, Xinya Wu, Pengfei Zhou, Richeng Xuan, Guang Liu, Xi Yang, Qiannan Zhu, and Hua Huang. Cmmu: A benchmark for chinese multi-modal multi-type question understanding and reasoning. *arXiv preprint arXiv:2401.14011*, 2024b.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Bohan Hou, Gen Li, Jindou Jia, Tuo An, Xinying Guo, Sicong Leng, Haoran Geng, Yanjie Ze, Tatsuya Harada, Philip Torr, et al. World model for robot learning: A comprehensive survey. *arXiv preprint arXiv:2605.00080*, 2026.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. Mlagentbench: Evaluating language agents on machine learning experimentation. *arXiv preprint arXiv:2310.03302*, 2023.

Igor Ivanov. Biolp-bench: Measuring understanding of biological lab protocols by large language models. *bioRxiv*, pp. 2024–08, 2024.

Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents. *Advances in Neural Information Processing Systems*, 37:10088–10116, 2024.

Ai Jian, Weijie Qiu, Xiaokun Wang, Peiyu Wang, Yunzhuo Hao, Jiangbo Pei, Yichen Wei, Yi Peng, and Xuchen Song. Csvqa: A chinese multimodal benchmark for evaluating stem reasoning capabilities of vlms. *arXiv preprint arXiv:2505.24120*, 2025.

Subbarao Kambhampati. Can large language models reason and plan? *Annals of the New York Academy of Sciences*, 1534(1):15–18, 2024.

Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Mudit Verma, Kaya Stechly, Siddhant Bhambri, Lucas Paul Saldyt, and Anil B Murthy. Position: Llms can't plan, but can help planning in llm-modulo frameworks. In *Forty-first International Conference on Machine Learning*, 2024.

Harsha Kokel, Michael Katz, Kavitha Srinivas, and Shirin Sohrabi. Acpbench: Reasoning about action, change, and planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 26559–26568, 2025.

Jakub Lála, Odhran O'Donoghue, Aleksandar Shtedritski, Sam Cox, Samuel G Rodriques, and Andrew D White. Paperqa: Retrieval-augmented generative agent for scientific research. *arXiv preprint arXiv:2312.07559*, 2023.

Zhiqian Lan, Yuxuan Jiang, Ruiqi Wang, Xuanbing Xie, Rongkui Zhang, Yicheng Zhu, Peihang Li, Tianshuo Yang, Tianxing Chen, Haoyu Gao, et al. Autobio: A simulation and benchmark for robotic automation in digital biology laboratory. *arXiv preprint arXiv:2505.14030*, 2025.

Jon M Laurent, Joseph D Janizek, Michael Ruzo, Michaela M Hinks, Michael J Hammerling, Siddharth Narayanan, Manvitha Ponnapati, Andrew D White, and Samuel G Rodriques. Lab-bench: Measuring capabilities of language models for biology research. *arXiv preprint arXiv:2407.10362*, 2024.

Jon M Laurent, Albert Bou, Michael Pieler, Conor Igoe, Alex Andonian, Siddharth Narayanan, James Braza, Alexandros Sanchez Vassopoulos, Jacob L Steenwyk, Blake Lash, et al. Labbench2: An improved benchmark for ai systems performing biology research. *arXiv preprint arXiv:2604.09554*, 2026.

Boyu Li, Chaoyi Xu, Haoqi Yuan, Xinrun Xu, Börje F Karlsson, Dongbin Zhao, Haoran Li, and Zongqing Lu. X-diffvla: X-embodied diffusion action heads for vision-language-action models. *arXiv preprint arXiv:2605.25044*, 2026a.

Haoming Li, Zhaoliang Chen, Jonathan Zhang, and Fei Liu. Planet: A collection of benchmarks for evaluating llms' planning capabilities. *arXiv preprint arXiv:2504.14773*, 2025.

Rui Li, Zixuan Hu, Wenxi Qu, Jinouwen Zhang, Zhenfei Yin, Sha Zhang, Xuantuo Huang, Hanqing Wang, Tai Wang, Jiangmiao Pang, et al. Labutopia: High-fidelity simulation and hierarchical benchmark for scientific embodied agents. *Advances in Neural Information Processing Systems*, 38, 2026b.

Fangru Lin, Emanuele La Malfa, Valentin Hofmann, Elle Michelle Yang, Anthony Cohn, and Janet B Pierrehumbert. Graph-enhanced large language models in asynchronous plan reasoning. *arXiv preprint arXiv:2402.02805*, 2024.

Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*, 2023.

Haowen Liu, Shaoxiong Yao, Haonan Chen, Jiawei Gao, Jiayuan Mao, Jia-Bin Huang, and Yilun Du. Simpact: Simulation-enabled action planning using vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20790–20801, 2026.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, volume 2024, pp. 52989–53046, 2024.

Yuyang Liu, Liuzhenghao Lv, Xiancheng Zhang, Jingya Wang Li Yuan, and Yonghong Tian. Bioprobench: Comprehensive dataset and benchmark in biological protocol understanding and reasoning. *arXiv preprint arXiv:2505.07889*, 2025.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024a.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521, 2022.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, volume 2024, pp. 23439–23554, 2024b.

Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature machine intelligence*, 6(5):525–535, 2024.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. Agentboard: An analytical evaluation board of multi-turn llm agents. *Advances in neural information processing systems*, 37:74325–74362, 2024.

MiniMax. MiniMax M3, 2026. Accessed 2026-06-03.

Arka Mukherjee and Shreya Ghosh. mmjee-eval: A bilingual multimodal benchmark for evaluating scientific reasoning in vision-language models. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 2268–2290, 2025.

Kushin Mukherjee, Donghao Ren, Dominik Moritz, and Yannick Assogba. Encqa: Benchmarking vision-language models on visual encodings for charts. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

OpenAI. Measuring ai's capability to accelerate biological research in the wet lab, December 2025.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. Spiqa: A dataset for multimodal question answering on scientific papers. *Advances in Neural Information Processing Systems*, 37:118807–118833, 2024.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Yusuf Roohani, Andrew Lee, Qian Huang, Jian Vora, Zachary Steinhart, Kexin Huang, Alexander Marson, Percy Liang, and Jure Leskovec. Biodiscoveryagent: An ai agent for designing genetic perturbation experiments. In *International Conference on Learning Representations*, volume 2025, pp. 26417–26466, 2025.

Jiacheng Ruan, Dan Jiang, Xian Gao, Ting Liu, Yuzhuo Fu, and Yangyang Kang. Mme-sci: A comprehensive and challenging science benchmark for multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 8760–8768, 2026.

Israfel Salazar, Manuel Fernández Burda, Shayekh Bin Islam, Arshia Soltani Moakhar, Shivalika Singh, Fabian Farestam, Angelika Romanou, Danylo Boiko, Dipika Khullar, Mike Zhang, et al. Kaleidoscope: In-language exams for massively multilingual vision evaluation. *arXiv preprint arXiv:2504.07072*, 2025.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: Evaluating ai's ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.

Kimi Team, Yifan Bai, Yiping Bao, Y Charles, Cheng Chen, Guanduo Chen, Haiting Chen, Huarong Chen, Jiahao Chen, Ningxin Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

MotuBrain Team, Chendong Xiang, Fan Bao, Haitian Liu, Hengkai Tan, Hongzhe Bi, James Li, Jiabao Liu, Jingrui Pang, Kiro Jing, et al. Motubrain: An advanced world action model for robot control. *arXiv preprint arXiv:2604.27792*, 2026.

Peiyan Tu, Hanxin Zhu, Jingwen Sun, Shaojie Ren, Cong Wang, Jiayi Luo, Xiaoqian Cheng, and Zhibo Chen. Embody4d: A generalist 4d world model for embodied ai. *arXiv preprint arXiv:2605.01799*, 2026.

Karthik Valmeekam, Matthew Marquez, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Planbench: An extensible benchmark for evaluating large language models on planning and reasoning about change. *Advances in Neural Information Processing Systems*, 36:38975–38987, 2023.

Karthik Valmeekam, Kaya Stechly, and Subbarao Kambhampati. Llms still can't plan; can lrms? a preliminary evaluation of openai's o1 on planbench. *arXiv preprint arXiv:2409.13373*, 2024.

Qiuyue Wang, Mingsheng Li, Jian Guan, Jinhui Ye, Sicheng Xie, Yitao Liu, Junhao Chen, Zhixuan Liang, Jie Zhang, Xintong Hu, et al. Qwen-vla: Unifying vision-language-action modeling across tasks, environments, and robot embodiments. *arXiv preprint arXiv:2605.30280*, 2026.

Weixuan Wang, Dongge Han, Daniel Madrigal Diaz, Jin Xu, Victor Rühle, and Saravan Rajmohan. Odysseybench: Evaluating llm agents on long-horizon complex office application workflows. *arXiv preprint arXiv:2508.09124*, 2025.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *Advances in Neural Information Processing Systems*, 37:113569–113697, 2024.

Hjalmar Wijk, Tao Lin, Joel Becker, Sami Jawhar, Neev Parikh, Thomas Broadley, Lawrence Chan, Michael Chen, Josh Clymer, Jai Dhyani, et al. Re-bench: Evaluating frontier ai r&d capabilities of language model agents against human experts. *arXiv preprint arXiv:2411.15114*, 2024.

xAI. xAI Docs: Models, 2026. Accessed 2026-06-02.

Xiaomi MiMo Team. Mimo-v2.5, 2026.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. Travelplanner: A benchmark for real-world planning with language agents. *arXiv preprint arXiv:2402.01622*, 2024a.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024b.

Wanghan Xu, Yuhao Zhou, Yifan Zhou, Qinglong Cao, Shuo Li, Jia Bu, Bo Liu, Yixin Chen, Xuming He, Xiangyu Zhao, et al. Probing scientific general intelligence of llms with scientist-aligned workflows. *arXiv preprint arXiv:2512.16969*, 2025.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *arXiv preprint arXiv:2504.08066*, 2025.

Han Yan, Zishang Xiang, Zeyu Zhang, and Hao Tang. Mwm: Mobile world models for action-conditioned consistent prediction. *arXiv preprint arXiv:2603.07799*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Xinwu Ye, Chengfan Li, Siming Chen, Wei Wei, and Robert Tang. Mmscibench: Benchmarking language models on chinese multimodal scientific problems. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 14621–14663, 2025.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, 2025.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Ge Zhang, Xinrun Du, Bei Chen, Yiming Liang, Tongxu Luo, Tianyu Zheng, Kang Zhu, Yuyang Cheng, Chunpu Xu, Shuyue Guo, et al. Cmmmu: A chinese massive multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2401.11944*, 2024.

Xiangyu Zhao, Wanghan Xu, Bo Liu, Yuhao Zhou, Fenghua Ling, Ben Fei, Xiaoyu Yue, Lei Bai, Wenlong Zhang, and Xiao-Ming Wu. Msearth: A multimodal scientific dataset and benchmark for phenomena uncovering in earth science. *arXiv preprint arXiv:2505.20740*, 2025.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 3: system demonstrations)*, pp. 400–410, 2024.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, volume 2024, pp. 15585–15606, 2024.

Max Zuo, Francisco Piedrahita Velez, Xiaochen Li, Michael Littman, and Stephen Bach. Planetarium: A rigorous benchmark for translating text to structured planning languages. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 11223–11240, 2025.

A Resource Comparison and Data Statistics

The supplementary material provides anonymized training and test data for both levels, including Level 1 asset images and the text inputs and reference outputs for both tasks.

Table 3: **Comparison with representative scientific multimodal and planning resources.** The columns correspond to the four challenges studied by LabHorizon. Green check marks denote full coverage, yellow triangles denote partial coverage, and red crosses denote dimensions outside the resource's primary design target.

Resource	Primary setting	Lab 3D Perception	Protocol Alignment	Long-Horizon Planning	Learning Loop
MSEarth (Zhao et al., 2025)	Earth-science figures and paper context	×	×	×	△
MME-SCI (Ruan et al., 2026)	Scientific multimodal reasoning	×	×	×	×
MMSciBench (Ye et al., 2025)	Chinese multimodal science problems	×	×	×	×
CSVQA (Jian et al., 2025)	STEM visual reasoning	×	×	×	×
SGI-WetExp. (Xu et al., 2025)	Wet experiment action pool	×	△	✓	×
BioProBench (Liu et al., 2025)	Biological protocol reasoning	×	✓	△	✓
LAB-Bench (Laurent et al., 2024)	Biology protocol troubleshooting	×	△	×	△
BioLP-bench (Ivanov, 2024)	Biological protocol mistake detection	×	△	×	×
ProtocolBench (Du et al., 2025)	Multi-agent communication protocols	×	×	△	△
SciGym (Duan et al., 2026)	Systems-biology dry lab	×	×	△	△
ACPBench (Kokel et al., 2025)	Formal action-change planning	×	×	△	×
Planetarium (Zuo et al., 2025)	NL-to-PDDL planning representation	×	×	△	△
LabHorizon	Laboratory assets, wet lab protocols	✓	✓	✓	✓

Table 4: **LabHorizon data statistics.** Level 1 covers multi-view laboratory asset perception and next-action prediction, while Level 2 covers wet experiment protocol-aligned long-horizon action sequence generation.

Level	Split	Samples	Assets	Protocols	Images / Steps
L1	Train	3,000	108	1,724	9,000 images
L1	Test	200	108	198	600 images
L2	Train	3,000	–	1,552	mean 12.61 steps, max 13
L2	Test	200	–	109	mean 12.57 steps, max 13
Total	–	6,400	108	2,342	–

B TRAINING HYPERPARAMETERS

Table 5: **Training hyperparameters for the Qwen3.6-35B-A3B domain model.** The table records the model, data mixture, LoRA setup, optimization configuration, runtime, and inference setting used for the current trained and trained+agents results.

Category	Hyperparameter	Value
Model	Base model	Qwen3.6-35B-A3B
Model	Model type	Hybrid MoE / DeltaNet, about 35B total and 3B active parameters
Data	Level 1 training examples	3,000
Data	Level 2 training examples	3,000
Data	Evaluation examples	200 Level 1 + 200 Level 2
Data	Data mixture ratio	1:1 Level 1 / Level 2
Data	Level 1 image setting	Three asset images per sample, max pixels 501760
Fine-tuning	Method	LoRA SFT via LLaMA-Factory
Fine-tuning	PEFT version	0.18.1
Fine-tuning	LoRA rank / alpha / dropout	32 / 64 / 0.10
Fine-tuning	LoRA target	All discovered linear modules in Qwen3.6 architecture
Fine-tuning	Vision tower / projector	Not frozen
Optimization	Learning rate	1.0×10^{-4}
Optimization	Scheduler	Cosine
Optimization	Warmup ratio	0.10
Optimization	Optimizer	LLaMA-Factory default AdamW-style optimizer
Optimization	NEFTune noise alpha	5
Batch	Per-device batch size	1
Batch	Gradient accumulation	4
Batch	Effective batch size	4 per device
Sequence	Cutoff length	4096
Runtime	Number of epochs	10
Runtime	Max training steps	2500
Runtime	Hardware	Single H100 NVL-class GPU in the migrated run
Runtime	Precision	bf16
Runtime	Gradient checkpointing	Enabled
Runtime	Seed	42
Runtime	Runtime	10014.77 s
Runtime	Train loss / eval loss	0.2691 / 0.4426
Inference	Decoding strategy	Greedy-style evaluation with temperature 0.0
Inference	Agent setting	direct / trained / trained+agents ablation

C CASE STUDY

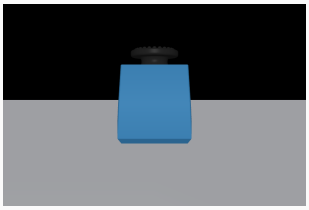
Each red case frame contains one complete evaluation sample in the public LabHorizon terminology. Level 1 cases are shown as multi-view asset inputs, historical actions, candidate next actions, gold reasoning, model reasoning, and evaluator scores. Level 2 cases are shown as context, goal, constraints, available inputs, action pool, gold action sequence, model action sequence, and evaluator scores. We include additional model outputs only when they are already available and do not require adding new biological procedure details solely for completeness. System prompts and implementation-only prompt wrappers are omitted because they are replaceable evaluation plumbing rather than dataset fields.

Level 1 Case Study: Reject-All Numerical Trap

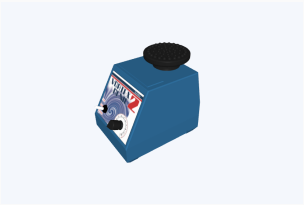
Asset

Asset name: Genie 2 Vortex Mixer. Asset family: vortex_mixer.

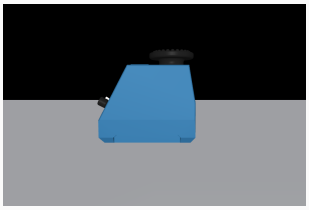
Visual Input



View 1



View 2



View 3

Historical Actions

1. Resin has already been combined with about 3.8 mL pooled eluate from three earlier affinity purifications.
2. Imidazole has been adjusted to a final concentration of 10 mM.
3. The mixture has been incubated at 4 C for 3 h at 10 rpm.
4. The resin has then been collected by centrifugation at 3000 x g for 2 min and washed three times in buffers containing 10 mM imidazole.

Candidate Next Actions

A. shake_sample
speed_rpm: 1000
temperature_c: 4
duration_min: 10
buffer: T150 + 200 mM imidazole
volume_ul: 120

B. shake_sample
speed_rpm: 1100
temperature_c: 4
duration_min: 8
buffer: T150 + 200 mM imidazole
volume_ul: 120

C. shake_sample
speed_rpm: 1100
temperature_c: 4
duration_min: 10
buffer: T150 + 300 mM imidazole
volume_ul: 120

D. incubate_binding
speed_rpm: 10
temperature_c: 4
duration_h: 4
substrate: Amylose beads + cell lysate

E. incubate_binding
speed_rpm: 12
temperature_c: 4
duration_h: 4
substrate: Amylose beads + cell lysate

F. **incubate_binding**
speed_rpm: 10
temperature_c: 4
duration_h: 3
substrate: Amylose beads + cell lysate

G. **collect_resin**
speed_xg: 3000
duration_min: 1
temperature_c: 4
transfer_tube_ml: 1.5

H. **collect_resin**
speed_xg: 3500
duration_min: 2
temperature_c: 4
transfer_tube_ml: 1.5

I. **collect_resin**
speed_xg: 3000
duration_min: 2
temperature_c: 4
transfer_tube_ml: 2.0

J. **none_of_the_above**

Gold Reasoning

1. The historical actions fix the local workflow stage after binding, collection, and three low-imidazole washes.
2. The true target family is A-C because the next elution step uses the pictured mixer and requires shaking at 1100 rpm and 4 C for 10 min with 120 uL T150 containing 200 mM imidazole.
3. A-C are each subtly wrong by one value: A changes 1100 rpm to 1000 rpm, B changes 10 min to 8 min, and C changes 200 mM imidazole to 300 mM.
4. D-F are realistic earlier binding incubations, and G-I are nearby collection steps with wrong parameters or transfer details.
5. Because every A-I candidate is plausible but incorrect, the valid next action is J.

Gold Next Action

J. **none_of_the_above**

Model Output: Qwen3.6-35B-A3B (trained)

1. The model localizes the stage after 10 mM imidazole binding, 4 C incubation, resin collection, and three washes.
2. It identifies the protocol target as first elution with 120 uL T150 + 200 mM imidazole at 4 C for 10 min at 1100 rpm.
3. It rejects A-C because each changes exactly one target parameter, and rejects D-I as earlier or nearby non-target actions.
4. Model next action: **J**.

Model Output: Claude Opus 4.7

1. Recorded model next action: **J**.

Evaluation

Gold next action: J
Qwen3.6-35B-A3B (trained) next action: J
Qwen3.6-35B-A3B (trained) Next Action Accuracy: 1 / 1
Claude Opus 4.7 next action: J
Claude Opus 4.7 Next Action Accuracy: 1 / 1

Level 1 Case Study: Asset-Protocol Alignment

Asset

Asset name: Beaker Family. Asset family: beaker.

Visual Input



View 1 View 2 View 3

Historical Actions

1. A rack with open 2 mL seed tubes has already been placed into the carrier on top of the stir plate inside the fume hood.
2. Seeds are ready for vapor-phase sterilization.
3. The pictured vessel holds a freshly prepared mixture made from 200 mL bleach and 5 mL concentrated HCl, with a stir bar inside, before the carrier is closed.

Candidate Next Actions

A. mix_solution
container_volume_ml: 250
bleach_ml: 200
hcl_ml: 5
stir_speed_rpm: 300-400
heat: off

B. mix_solution
container_volume_ml: 250
bleach_ml: 210
hcl_ml: 5
stir_speed_rpm: 300-400
heat: off

C. mix_solution
container_volume_ml: 250
bleach_ml: 200
hcl_ml: 4
stir_speed_rpm: 300-400
heat: off

D. vent_seeds
duration_min: 15-20

lid_position: slightly offset
location: fume hood

E. vent_seeds
duration_min: 10-15
lid_position: slightly offset
location: fume hood

F. vent_seeds
duration_min: 15-20
lid_position: fully closed
location: fume hood

G. place_seed_tubes
tube_volume_ml: 2
tube_state: open
support: rack

H. place_seed_tubes
tube_volume_ml: 1.5
tube_state: open
support: rack

I. place_seed_tubes
tube_volume_ml: 2
tube_state: closed
support: rack

J. none_of_the_above

Gold Reasoning

1. The current stage is immediately after the open seed tubes have been placed into the carrier.
2. The protocol requires a 250 mL vessel containing 200 mL bleach and 5 mL concentrated HCl with a stir bar.
3. The operation should start stirring at 300-400 rpm with heat off.
4. A-C form the target family; only A preserves all numerical values and heat state.
5. D-F are later venting actions, and G-I are earlier seed-tube placement actions.

Gold Next Action

A. mix_solution
container_volume_ml: 250
bleach_ml: 200
hcl_ml: 5
stir_speed_rpm: 300-400
heat: off

Model Output: Qwen3.6-35B-A3B (trained)

1. The model identifies that the seed tubes have already been placed and that the next action concerns the bleach/HCl vessel.
2. It compares A-C as the same mixing action family and finds only A keeps 250 mL container, 200 mL bleach, 5 mL HCl, 300-400 rpm, and heat off.
3. It rejects D-F as later venting and G-I as earlier placement.

4. Model next action: **A**.

Model Output: Claude Opus 4.7

1. Recorded model next action: **A**.

Evaluation

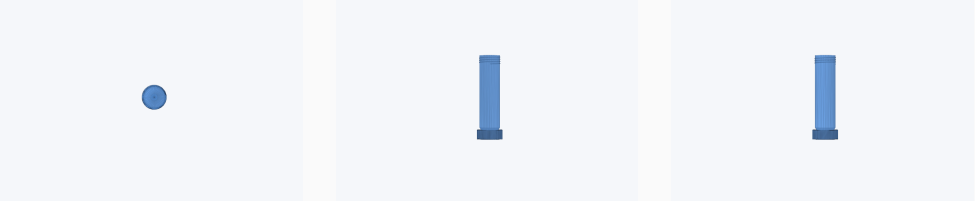
Gold next action: A
Qwen3.6-35B-A3B (trained) next action: A
Qwen3.6-35B-A3B (trained) Next Action Accuracy: 1 / 1
Claude Opus 4.7 next action: A
Claude Opus 4.7 Next Action Accuracy: 1 / 1

Level 1 Case Study: Centrifugation Parameter Trap

Asset

Asset name: 50 mL Screw-Cap Centrifuge Tube. Asset family: centrifuge_tube_50ml.

Visual Input



View 1 View 2 View 3

Historical Actions

1. Crude extract has already been transferred into the pictured centrifuge tube.
2. It has been centrifuged at 4 C and 1,000 x g for 15 min.
3. The supernatant has been collected while the pellet was discarded.
4. The supernatant has now been moved into a new pictured tube for the second round.

Candidate Next Actions

A. centrifuge_sample
speed_xg: 2500
duration_min: 10
temperature_c: 4

B. centrifuge_sample
speed_xg: 2500
duration_min: 15
temperature_c: 4

C. centrifuge_sample
speed_xg: 3000
duration_min: 15
temperature_c: 4

D. transfer_supernatant
container_volume_ml: 50
destination: new container

pellet_action: discard

E. transfer_supernatant
container_volume_ml: 40
destination: new container
pellet_action: discard

F. transfer_supernatant
container_volume_ml: 50
destination: same container
pellet_action: discard

G. filter_sample
filter_um: 0.22
syringe_volume_ml: 10
needle: without

H. filter_sample
filter_um: 0.45
syringe_volume_ml: 10
needle: without

I. filter_sample
filter_um: 0.22
syringe_volume_ml: 5
needle: without

J. none_of_the_above

Gold Reasoning

1. The first transfer, first 1,000 x g spin, pellet discard, and supernatant transfer are complete.
2. The next protocol step is the second centrifugation at 4 C, 2,500 x g, and 15 min.
3. A-C form the target centrifugation family; B exactly matches the required values.
4. A changes duration to 10 min and C changes speed to 3,000 x g.
5. D-F are nearby transfer actions and G-I are later filtration actions, not the current operation.

Gold Next Action

B. centrifuge_sample
speed_xg: 2500
duration_min: 15
temperature_c: 4

Model Output: Qwen3.6-35B-A3B (trained)

1. The model correctly localizes the workflow after the first 1,000 x g spin and supernatant transfer.
2. It identifies the second-round operation as 2,500 x g for 15 min at 4 C.
3. It rejects the duration and speed perturbations in A and C.
4. Model next action: **B**.

Model Output: Claude Opus 4.7

1. Recorded model next action: **B**.

Evaluation

Gold next action: B
Qwen3.6-35B-A3B (trained) next action: B
Qwen3.6-35B-A3B (trained) Next Action Accuracy: 1 / 1
Claude Opus 4.7 next action: B
Claude Opus 4.7 Next Action Accuracy: 1 / 1

Level 2 Case Study: RNA-Extraction Cleanup

Context

The stage is late RNA extraction after precipitation, where a wastewater RNA pellet must be washed cold, briefly dried, resuspended, and recorded for downstream SARS-CoV-2 N1 and PMMoV assays.

Goal

Process the centrifuged wastewater RNA pellet through cold ethanol washing and recovery, then prepare the recovered RNA for downstream RT-qPCR while preserving pellet integrity.

Constraints

1. Begin from a wastewater RNA pellet that requires cold centrifugation-based cleanup.
2. Use 12,000 x g at 4 C, with one 10 min recovery spin and two 3 min wash spins.
3. Maintain cold handling; use two rounds of ice-cold 75% ethanol without aggressive mixing.
4. Remove residual liquid before drying; air-dry on ice for only 5-10 min.
5. Resuspend in exactly 40 uL DEPC-treated water.
6. Record 18 uL for SARS-CoV-2 N1 triplicate runs and 9 uL for PMMoV triplicate runs.
7. Preserve one dependency chain from the original pellet to the final documented RNA state.

Available Inputs

rna_pellet_tube: wastewater RNA pellet in a 50 mL tube after precipitation and before recovery cleanup.

ethanol_wash_1: first freshly prepared ice-cold 75% ethanol wash reagent.

ethanol_wash_2: second freshly prepared ice-cold 75% ethanol wash reagent.

depc_water: DEPC-treated nuclease-free water for final RNA resuspension.

Action Pool

centrifuge_sample *Spins a sample at a specified relative centrifugal force, duration, and temperature.*
Inputs: sample, speed_xg, duration_min, temperature_c
Output: centrifuged sample handle

incubate_sample *Holds a sample under a defined temperature and condition for a fixed duration.*
Inputs: sample, temperature_c, duration_min, condition
Output: incubated sample handle

heat_sample *Applies controlled heating to a sample with the specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: heated sample handle

cool_sample *Chills or holds a sample at a target low temperature with a specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: cooled sample handle

set_thermal_mixer *Mixes a sample while controlling temperature, shaking speed, and duration.*
Inputs: sample, temperature_c, speed_rpm, duration_min

Output: thermomixed sample handle

run_thermal_cycler *Runs a named thermal-cycler program with cycle count and lid temperature.*
Inputs: plate, program_name, cycles, lid_temperature_c
Output: thermocycled product handle

dry_sample *Removes residual liquid using a specified drying method and duration.*
Inputs: sample, method, temperature_c, duration_min
Output: dried sample handle

store_sample *Places a sample in a defined storage condition for later handling.*
Inputs: sample, temperature_c, duration_h, container
Output: stored sample handle

measure_signal *Reads an instrument signal from a sample at a specified wavelength.*
Inputs: sample, signal_name, wavelength_nm, device
Output: signal measurement handle

record_measurement *Records an assay metric, unit, or aliquot note for a sample.*
Inputs: sample, metric, unit, note
Output: measurement record handle

mix_sample *Combines or resuspends material using a specified method, time, and speed.*
Inputs: sample, method, duration_s, speed
Output: mixed sample handle

vortex_mix *Mixes a sample on a vortex with controlled speed, temperature, and time.*
Inputs: sample, duration_s, speed_rpm, temperature_c
Output: vortexed sample handle

stir_with_rod *Stirs material in a container using a manual rod-like mixing step.*
Inputs: container, duration_s, speed_desc, temperature_c
Output: stirred sample handle

shake_container *Shakes a container with specified amplitude, repetitions, and duration.*
Inputs: sample, amplitude_mm, repeat_n, duration_s
Output: shaken sample handle

Gold Action Sequence

1. **centrifuge_sample**
   Input: rna_pellet_tube
   Parameters: 12,000 x g; 10 min; 4 C
   Output: pellet_recovered
2. **mix_sample**
   Input: pellet_recovered
   Parameters: method = ethanol_wash_1; duration = 5 s; speed = slow side-wall addition
   Output: wash1_added
3. **centrifuge_sample**
   Input: wash1_added
   Parameters: 12,000 x g; 3 min; 4 C
   Output: wash1_spun
4. **mix_sample**
   Input: wash1_spun
   Parameters: method = ethanol_wash_2; duration = 5 s; speed = slow side-wall addition
   Output: wash2_added

5. **centrifuge_sample**
   Input: wash2_added
   Parameters: 12,000 x g; 3 min; 4 C
   Output: wash2_spun
6. **cool_sample**
   Input: wash2_spun
   Parameters: 0 C; 1 min; device = ice
   Output: pellet_chilled
7. **dry_sample**
   Input: pellet_chilled
   Parameters: method = air dry on ice; 0 C; 7 min
   Output: pellet_dried
8. **mix_sample**
   Input: pellet_dried
   Parameters: method = depc_water; duration = 10 s; speed = gentle resuspension to 40 uL final volume
   Output: rna_resuspended
9. **record_measurement**
   Input: rna_resuspended
   Parameters: metric = SARS-CoV-2 N1 aliquot volume; unit = uL; note = 18 uL reserved for triplicate RT-qPCR
   Output: n1_aliquot_record
10. **record_measurement**
    Input: rna_resuspended
    Parameters: metric = PMMoV aliquot volume; unit = uL; note = 9 uL reserved for triplicate RT-qPCR
    Output: pmmov_aliquot_record

Model Action Sequence: Qwen3.6-35B-A3B (trained)

1. **centrifuge_sample**
   Input: rna_pellet_tube
   Parameters: 12,000 x g; 10 min; 4 C
   Output: step1
2. **mix_sample**
   Input: ethanol_wash_1
   Parameters: method = inversion; duration = 10 s; speed = gentle
   Output: step2
   Error: The reagent is mixed by itself rather than being added to the pellet chain.
3. **mix_sample**
   Input: step1
   Parameters: method = pipetting; duration = 10 s; speed = gentle
   Output: step3
4. **centrifuge_sample**
   Input: step3
   Parameters: 12,000 x g; 3 min; 4 C
   Output: step4
5. **mix_sample**
   Input: ethanol_wash_2
   Parameters: method = inversion; duration = 10 s; speed = gentle
   Output: step5
   Error: The second wash reagent is again processed outside the pellet dependency chain.

6. **mix_sample**
   Input: step4
   Parameters: method = pipetting; duration = 10 s; speed = gentle
   Output: step6
7. **centrifuge_sample**
   Input: step6
   Parameters: 12,000 x g; 3 min; 4 C
   Output: step7
8. **dry_sample**
   Input: step7
   Parameters: method = air dry on ice; 0 C; 7 min
   Output: step8
9. **mix_sample**
   Input: depc_water
   Parameters: method = inversion; duration = 10 s; speed = gentle
   Output: step9
   Error: Water is mixed as a standalone sample instead of resuspending the pellet.
10. **mix_sample**
    Input: step8
    Parameters: method = pipetting; duration = 10 s; speed = gentle
    Output: step10
    Error: The exact 40 uL resuspension volume is not preserved.
11. **record_measurement**
    Input: step10
    Parameters: metric = SARS-CoV-2 N1 aliquot volume; unit = uL; note = 18 uL for triplicate RT-qPCR
    Output: step11
12. **record_measurement**
    Input: step10
    Parameters: metric = PMMoV aliquot volume; unit = uL; note = 9 uL for triplicate RT-qPCR
    Output: step12

Evaluation

Qwen3.6-35B-A3B (trained) Level 2 Action Sequence Similarity: 0.1667
Qwen3.6-35B-A3B (trained) Level 2 Parameter Accuracy: 0.3462
Qwen3.6-35B-A3B (trained) Level 2 Final Score: 0.2564
Observed error pattern: The output preserves some cleanup actions but breaks the pellet dependency chain during wash handling and misses exact resuspension accounting.

Level 2 Case Study: Thermal-Cycler Dependency Failure

Context

A pooled sgRNA library has already been amplified and purified; the current stage processes a Golden Gate assembly reaction in a thermal cycler and leaves the assembled plasmid library in the correct post-run state.

Goal

Execute the thermal-cycling portion of Golden Gate assembly for sgRNA library plasmid cloning and leave the assembled product cold for subsequent purification.

Constraints

1. Start from the Golden Gate assembly reaction and stay centered on the thermal cycler stage.

2. Capture 20 cycles of 37 C for 5 min and 16 C for 5 min, followed by a final 37 C incubation for 5 min.
3. Seal the reaction vessel before thermocycling.
4. Open the thermal cycler lid before loading and close it before the run.
5. Establish a post-run 4 C hold or equivalent cooled state.
6. Use only action-pool actions and preserve dependency chains from earlier outputs.
7. Include repeated action families with small numeric variations.

Available Inputs

golden_gate_reaction: prepared Golden Gate reaction with vector, purified sgRNA PCR inserts, enzymes, buffer, ATP, DTT, and water.

gg_plate: plate or tube strip holding Golden Gate reactions.

thermal_cycler: thermal cycler used to run the assembly program.

cold_block: cooling device used to maintain the reaction at 4 C after cycling.

Action Pool

centrifuge_sample *Spins a sample at a specified relative centrifugal force, duration, and temperature.*
Inputs: sample, speed_xg, duration_min, temperature_c
Output: centrifuged sample handle

open_device *Opens an instrument or device target in a specified mode.*
Inputs: device, target, mode
Output: opened device-state handle

close_device *Closes an instrument or device target in a specified mode.*
Inputs: device, target, mode
Output: closed device-state handle

press_button *Activates a named device button for a specified hold time and repeat count.*
Inputs: device, button, hold_s, repeat_n
Output: device state after button activation

open_thermal_cycler_lid *Opens the thermal-cycler lid for loading or unloading a plate.*
Inputs: instrument, latch_state, plate
Output: open thermal-cycler state handle

close_thermal_cycler_lid *Closes and secures the thermal-cycler lid before running a program.*
Inputs: instrument, latch_state, plate
Output: closed thermal-cycler state handle

seal_plate *Seals a plate or tube strip with specified material, pressure, and time.*
Inputs: plate, seal_type, pressure, duration_s
Output: sealed plate handle

incubate_sample *Holds a sample under a defined temperature and condition for a fixed duration.*
Inputs: sample, temperature_c, duration_min, condition
Output: incubated sample handle

heat_sample *Applies controlled heating to a sample with the specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: heated sample handle

cool_sample *Chills or holds a sample at a target low temperature with a specified device.*
Inputs: sample, temperature_c, duration_min, device

Output: cooled sample handle

set_thermal_mixer *Mixes a sample while controlling temperature, shaking speed, and duration.*
Inputs: sample, temperature_c, speed_rpm, duration_min
Output: thermomixed sample handle

run_thermal_cycler *Runs a named thermal-cycler program with cycle count and lid temperature.*
Inputs: plate, program_name, cycles, lid_temperature_c
Output: thermocycled product handle

dry_sample *Removes residual liquid using a specified drying method and duration.*
Inputs: sample, method, temperature_c, duration_min
Output: dried sample handle

store_sample *Places a sample in a defined storage condition for later handling.*
Inputs: sample, temperature_c, duration_h, container
Output: stored sample handle

Gold Action Sequence

1. **seal_plate**
   Input: gg_plate
   Parameters: adhesive film; firm pressure; 5 s
   Output: sealed_gg_plate
2. **open_thermal_cycler_lid**
   Input: thermal_cycler; sealed_gg_plate
   Parameters: latch = unlocked
   Output: opened_tc
3. **cool_sample**
   Input: sealed_gg_plate
   Parameters: 4 C; 2 min; device = cold_block
   Output: cooled_loaded_plate
4. **close_thermal_cycler_lid**
   Input: thermal_cycler; cooled_loaded_plate
   Parameters: latch = locked
   Output: closed_tc
5. **run_thermal_cycler**
   Input: cooled_loaded_plate
   Parameters: program = Golden_Gate_37C_16C_20x_final_37C; cycles = 20; lid = 105 C
   Output: thermocycled_assembly
6. **cool_sample**
   Input: thermocycled_assembly
   Parameters: 4 C; 5 min; device = thermal cycler
   Output: postrun_cool
7. **open_thermal_cycler_lid**
   Input: thermal_cycler; postrun_cool
   Parameters: latch = unlocked
   Output: opened_tc_after_run
8. **store_sample**
   Input: postrun_cool
   Parameters: 4 C; 1 h; container = thermal_cycler_hold
   Output: held_product

Model Action Sequence: Qwen3.6-35B-A3B (trained)

1. **open_thermal_cycler_lid**
   Input: thermal_cycler; raw gg_plate
   Parameters: latch = open
   Output: opened_tc
   Error: Opens before sealing and uses a non-pool latch value.
2. **seal_plate**
   Input: gg_plate
   Parameters: adhesive film; firm pressure; 5 s
   Output: sealed_plate
3. **close_thermal_cycler_lid**
   Input: thermal_cycler; sealed_plate
   Parameters: latch = closed
   Output: closed_tc
   Error: Latch should be locked and the cold loading stage is missing.
4. **run_thermal_cycler**
   Input: sealed_plate
   Parameters: program = Golden Gate assembly 37 C / 16 C; cycles = 20; lid = 105 C
   Output: thermal_cycled_product
5. **incubate_sample**
   Input: thermal_cycled_product
   Parameters: 37 C; 5 min; condition = static
   Output: incubation_37c
6. **incubate_sample**
   Input: incubation_37c
   Parameters: 16 C; 5 min; condition = static
   Output: incubation_16c
7. **incubate_sample**
   Input: incubation_16c
   Parameters: 37 C; 5 min; condition = static
   Output: incubation_final
   Error: The model expands thermal-cycler internal stages as separate incubations after the run.
8. **cool_sample**
   Input: incubation_final
   Parameters: 4 C; 1 min; device = cold_block
   Output: cooled_sample
9. **store_sample**
   Input: cooled_sample
   Parameters: 4 C; 1 h; container = cold_block
   Output: stored_sample

Evaluation

Qwen3.6-35B-A3B (trained) Level 2 Action Sequence Similarity: 0.0000
Qwen3.6-35B-A3B (trained) Level 2 Parameter Accuracy: 0.1778
Qwen3.6-35B-A3B (trained) Level 2 Final Score: 0.0889
Observed error pattern: The output is parseable but reverses lid-sealing order, runs the thermal program on the wrong dependency, and expands a device program into local incubation calls.

Level 2 Case Study: Long-Horizon Success

Context

Harvested organoids have been separated into a cell-count tube and an infection tube; the workflow requires medium exchange, pelleting, dissociation, incubation, cooling, and virus aliquot preparation.

Goal

Process the organoid harvest so that the cell-count fraction is enzymatically dissociated and washed, the infection fraction is mechanically resuspended and chilled, and frozen virus aliquots are thawed on ice.

Constraints

1. Use one tube for cell counting and one tube for infection setup.
2. Both branches require an initial 100 x g, 5 min, 4 C pelleting spin.
3. The cell-count branch receives 1000 uL TrypLE Select and incubates at 37 C for 5 min.
4. The infection branch receives 2000 uL cold basal medium and remains on ice.
5. Adjust the cell-count branch to 13 mL total volume and the infection branch to 10 mL using cold basal medium.
6. Both branches require a second 100 x g, 5 min, 4 C spin.
7. Virus aliquots from -80 C storage should thaw on ice.

Available Inputs

tube1_pellet: harvested organoid pellet reserved for cell counting.

tube2_pellet: harvested organoid pellet reserved for infection setup.

basal_medium_cold: cold basal medium for washes and volume adjustment.

tryple_select: TrypLE Select reagent for enzymatic dissociation.

virus_aliquots_frozen: ready-to-use virus aliquots stored at -80 C.

Action Pool

centrifuge_sample *Spins a sample at a specified relative centrifugal force, duration, and temperature.*
Inputs: sample, speed_xg, duration_min, temperature_c
Output: centrifuged sample handle

incubate_sample *Holds a sample under a defined temperature and condition for a fixed duration.*
Inputs: sample, temperature_c, duration_min, condition
Output: incubated sample handle

heat_sample *Applies controlled heating to a sample with the specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: heated sample handle

cool_sample *Chills or holds a sample at a target low temperature with a specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: cooled sample handle

set_thermal_mixer *Mixes a sample while controlling temperature, shaking speed, and duration.*
Inputs: sample, temperature_c, speed_rpm, duration_min
Output: thermomixed sample handle

run_thermal_cycler *Runs a named thermal-cycler program with cycle count and lid temperature.*
Inputs: plate, program_name, cycles, lid_temperature_c
Output: thermocycled product handle

dry_sample *Removes residual liquid using a specified drying method and duration.*
Inputs: sample, method, temperature_c, duration_min
Output: dried sample handle

store_sample *Places a sample in a defined storage condition for later handling.*
Inputs: sample, temperature_c, duration_h, container
Output: stored sample handle

filter_sample *Passes a sample through a defined filter into a destination vessel.*
Inputs: sample, filter_type, pore_size_um, destination
Output: filtered sample handle

magnetic_separate *Separates magnetic material on a rack and selects the target phase.*
Inputs: sample, rack, duration_min, collect_phase
Output: magnetically separated sample handle

place_on_magnet *Places a sample on a magnetic rack for bead capture or separation.*
Inputs: sample, rack, duration_min, orientation
Output: sample handle after magnetic placement

collect_fraction *Collects a specified phase and volume into a destination vessel.*
Inputs: sample, phase, destination, volume_ul
Output: collected fraction handle

add_reagent *Adds a reagent to a sample with a specified volume and mixing mode.*
Inputs: sample, reagent, volume_ul, mixing
Output: updated sample handle

aliquot_sample *Splits a sample into destination aliquots with defined volume and count.*
Inputs: sample, destination, volume_ul, aliquot_count
Output: aliquoted sample handle

Gold Action Sequence

1. **centrifuge_sample**
   Input: tube1_pellet
   Parameters: 100 x g; 5 min; 4 C
   Output: tube1_spin1
2. **centrifuge_sample**
   Input: tube2_pellet
   Parameters: 100 x g; 5 min; 4 C
   Output: tube2_spin1
3. **add_reagent**
   Input: tube1_spin1
   Parameters: reagent = tryple_select; volume = 1000 uL; mixing = pipette mixing
   Output: tube1_tryple
4. **incubate_sample**
   Input: tube1_tryple
   Parameters: 37 C; 5 min; condition = static
   Output: tube1_dissociated
5. **add_reagent**
   Input: tube2_spin1
   Parameters: reagent = basal_medium_cold; volume = 2000 uL; mixing = pipette mixing
   Output: tube2_resuspended
6. **cool_sample**
   Input: tube2_resuspended

Parameters: 0 C; 5 min; device = ice
Output: tube2_chilled

7. **add_reagent**
   Input: tube1_dissociated
   Parameters: reagent = basal_medium_cold; volume = 12000 uL; mixing = gentle inversion
   Output: tube1_brought_to_volume
8. **add_reagent**
   Input: tube2_chilled
   Parameters: reagent = basal_medium_cold; volume = 8000 uL; mixing = gentle inversion
   Output: tube2_brought_to_volume
9. **centrifuge_sample**
   Input: tube1_brought_to_volume
   Parameters: 100 x g; 5 min; 4 C
   Output: tube1_spin2
10. **centrifuge_sample**
    Input: tube2_brought_to_volume
    Parameters: 100 x g; 5 min; 4 C
    Output: tube2_spin2
11. **cool_sample**
    Input: virus_aliquots_frozen
    Parameters: 0 C; 5 min; device = ice
    Output: virus_aliquots_thawed

Model Action Sequence: Qwen3.6-35B-A3B (trained)

1. **centrifuge_sample**
   Input: tube1_pellet
   Parameters: 100 x g; 5 min; 4 C
   Output: step1
2. **centrifuge_sample**
   Input: tube2_pellet
   Parameters: 100 x g; 5 min; 4 C
   Output: step2
3. **add_reagent**
   Input: step1
   Parameters: reagent = tryple_select; volume = 1000 uL; mixing = resuspension mixing
   Output: step3
4. **incubate_sample**
   Input: step3
   Parameters: 37 C; 5 min; condition = static
   Output: step4
5. **add_reagent**
   Input: step2
   Parameters: reagent = basal_medium_cold; volume = 2000 uL; mixing = resuspension mixing
   Output: step5
6. **cool_sample**
   Input: step5
   Parameters: 0 C; 5 min; device = ice
   Output: step6
7. **add_reagent**
   Input: step4
   Parameters: reagent = basal_medium_cold; volume = 12000 uL; mixing = generic mixing
   Output: step7

8. **add_reagent**
   Input: step6
   Parameters: reagent = basal_medium_cold; volume = 8000 uL; mixing = generic mixing
   Output: step8
9. **centrifuge_sample**
   Input: step7
   Parameters: 100 x g; 5 min; 4 C
   Output: step9
10. **centrifuge_sample**
   Input: step8
   Parameters: 100 x g; 5 min; 4 C
   Output: step10
11. **cool_sample**
   Input: virus_aliquots_frozen
   Parameters: 0 C; 5 min; device = ice
   Output: step11

Evaluation

Qwen3.6-35B-A3B (trained) Level 2 Action Sequence Similarity: 1.0000
Qwen3.6-35B-A3B (trained) Level 2 Parameter Accuracy: 0.9091
Qwen3.6-35B-A3B (trained) Level 2 Final Score: 0.9545
Observed success pattern: The output preserves both experimental branches, keeps the major centrifugation and cooling dependencies, and only leaves minor parameter-level mismatches.

Level 2 Case Study: Tissue-Dissociation Planning

Context

Embryonic cortical tissue is in a microtube with cold L15 medium and must be enzymatically dissociated, pelleted, resuspended, chilled, embedded in Matrigel, and cultured.

Goal

Process cortical tissue through digestion, quenching, low-speed pelleting, chilled resuspension, Matrigel handling, and final culture incubation so that a Matrigel-embedded neuronal aggregate is ready for the initial 3-day culture period.

Constraints

1. Begin from cortex tissue in ice-cold L15 medium and include a brief sedimenting spin.
2. Represent 0.05% Trypsin-EDTA digestion at 37 C for 25 min, with mixing during digestion.
3. Quench with room-temperature FBS before removing supernatant.
4. Model mechanical dissociation by pipetting to homogeneity.
5. Perform gentle centrifugation at 80 x g for 5 min before chilled L15 resuspension.
6. Preserve cold post-pellet L15 handling and Matrigel handling, including 1 min ice adherence.
7. Include Matrigel gelation for 25 min at 37 C and finish with 3 days of culture at 37 C under humidified 5% CO2-like conditions.

Available Inputs

cortical_tissue_tube: 1.5 mL tube with minced cortices from four embryos in 200 uL ice-cold L15 medium.

trypsin_edta_005: ice-cold 0.05% Trypsin-EDTA diluted in L15.

fbs_rt: room-temperature fetal bovine serum for quenching digestion.

dissociation_buffer: buffer used for trituration into a homogeneous neuronal suspension.

l15_cold: ice-cold L15 medium for chilled resuspension.
matrigel_mix: 75% Matrigel in L15 for embedding.
dmem_prewarmed: prewarmed DMEM culture medium for post-gelation culture.

Action Pool

centrifuge_sample *Spins a sample at a specified relative centrifugal force, duration, and temperature.*
Inputs: sample, speed_xg, duration_min, temperature_c
Output: centrifuged sample handle

incubate_sample *Holds a sample under a defined temperature and condition for a fixed duration.*
Inputs: sample, temperature_c, duration_min, condition
Output: incubated sample handle

heat_sample *Applies controlled heating to a sample with the specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: heated sample handle

cool_sample *Chills or holds a sample at a target low temperature with a specified device.*
Inputs: sample, temperature_c, duration_min, device
Output: cooled sample handle

set_thermal_mixer *Mixes a sample while controlling temperature, shaking speed, and duration.*
Inputs: sample, temperature_c, speed_rpm, duration_min
Output: thermomixed sample handle

run_thermal_cycler *Runs a named thermal-cycler program with cycle count and lid temperature.*
Inputs: plate, program_name, cycles, lid_temperature_c
Output: thermocycled product handle

dry_sample *Removes residual liquid using a specified drying method and duration.*
Inputs: sample, method, temperature_c, duration_min
Output: dried sample handle

store_sample *Places a sample in a defined storage condition for later handling.*
Inputs: sample, temperature_c, duration_h, container
Output: stored sample handle

mix_sample *Combines or resuspends material using a specified method, time, and speed.*
Inputs: sample, method, duration_s, speed
Output: mixed sample handle

vortex_mix *Mixes a sample on a vortex with controlled speed, temperature, and time.*
Inputs: sample, duration_s, speed_rpm, temperature_c
Output: vortexed sample handle

stir_with_rod *Stirs material in a container using a manual rod-like mixing step.*
Inputs: container, duration_s, speed_desc, temperature_c
Output: stirred sample handle

shake_container *Shakes a container with specified amplitude, repetitions, and duration.*
Inputs: sample, amplitude_mm, repeat_n, duration_s
Output: shaken sample handle

filter_sample *Passes a sample through a defined filter into a destination vessel.*
Inputs: sample, filter_type, pore_size_um, destination
Output: filtered sample handle

magnetic_separate *Separates magnetic material on a rack and selects the target phase.*
Inputs: sample, rack, duration_min, collect_phase
Output: magnetically separated sample handle

Gold Action Sequence

1. **centrifuge_sample**
   Input: cortical_tissue_tube
   Parameters: 2,500 x g; 0.5 min; 4 C
   Output: brief_spin
2. **incubate_sample**
   Input: brief_spin
   Parameters: 37 C; 25 min; condition = static Trypsin-EDTA digestion
   Output: trypsin_digest
3. **mix_sample**
   Input: trypsin_digest
   Parameters: method = gentle pipetting; duration = 10 s; speed = gentle
   Output: digest_mix
4. **centrifuge_sample**
   Input: digest_mix
   Parameters: 2,500 x g; 0.5 min; 22 C
   Output: quench_spin
5. **mix_sample**
   Input: quench_spin
   Parameters: method = moderate pipetting with FBS quench; duration = 30 s; speed = moderate
   Output: dissociated_cells
6. **centrifuge_sample**
   Input: dissociated_cells
   Parameters: 80 x g; 5 min; 22 C
   Output: gentle_pellet
7. **cool_sample**
   Input: gentle_pellet
   Parameters: 0 C; 1 min; device = ice bath
   Output: chilled_resuspension
8. **mix_sample**
   Input: chilled_resuspension
   Parameters: method = gentle pipetting in cold L15; duration = 15 s; speed = gentle
   Output: resuspended_pellet
9. **cool_sample**
   Input: matrigel_mix
   Parameters: 0 C; 1 min; device = ice bath
   Output: matrigel_chilled
10. **incubate_sample**
    Input: resuspended_pellet; matrigel_chilled
    Parameters: 0 C; 1 min; condition = ice adherence with chilled Matrigel
    Output: ice_adhesion
11. **incubate_sample**
    Input: ice_adhesion
    Parameters: 37 C; 25 min; condition = humidified 5% CO2
    Output: gelated_embed

12. **incubate_sample**
    Input: gelated_embed
    Parameters: 37 C; 72 h; condition = humidified 5% CO2 with prewarmed DMEM
    Output: initial_culture

Model Action Sequence: Qwen3.6-35B-A3B (trained)

1. **centrifuge_sample**
   Input: cortical_tissue_tube
   Parameters: 80 x g; 0.5 min; 4 C
   Output: step1
   Error: First spin speed should be 2,500 x g.
2. **mix_sample**
   Input: step1
   Parameters: method = gentle pipetting; duration = 10 s; speed = gentle
   Output: step2
   Error: Digestion incubation is skipped before this mixing step.
3. **incubate_sample**
   Input: step2
   Parameters: 37 C; 25 min; condition = static
   Output: step3
4. **mix_sample**
   Input: step3
   Parameters: method = gentle pipetting; duration = 10 s; speed = gentle
   Output: step4
5. **mix_sample**
   Input: step4
   Parameters: method = gentle pipetting; duration = 10 s; speed = gentle
   Output: step5
   Error: Quench spin and moderate dissociation are collapsed into repeated generic mixing.
6. **centrifuge_sample**
   Input: step5
   Parameters: 80 x g; 5 min; 4 C
   Output: step6
   Error: Temperature differs from the 22 C low-speed pelleting stage.
7. **mix_sample**
   Input: step6
   Parameters: method = gentle pipetting; duration = 10 s; speed = gentle
   Output: step7
8. **cool_sample**
   Input: step7
   Parameters: 0 C; 1 min; device = ice
   Output: step8
9. **incubate_sample**
   Input: step8
   Parameters: 37 C; 25 min; condition = static
   Output: step9
10. **incubate_sample**
    Input: step9
    Parameters: 37 C; 4320 min; condition = humidified 5% CO2
    Output: step10
    Error: 4320 min represents 3 days but skips explicit Matrigel chilling and ice-adherence dependencies.

Evaluation

Qwen3.6-35B-A3B (trained) Level 2 Action Sequence Similarity: 0.3333
Qwen3.6-35B-A3B (trained) Level 2 Parameter Accuracy: 0.4038
Qwen3.6-35B-A3B (trained) Level 2 Final Score: 0.3686
Observed error pattern: The model recovers the digestion-to-culture theme but collapses several local stages and loses explicit Matrigel handling dependencies.

Level 2 Case Study: Hydrogel Clone Cleanup

Context

Hydrogel-grown clonal spheres have already been collected after several days of expansion, and the workflow now shifts to recovering intact clones for downstream transcriptome barcoding. The immediate task is to digest the PEGDT-MALDEX material, clear remaining gel debris, and gently clean the clone suspension because the settled clone fraction is fragile and not tightly packed.

Goal

Generate a cleaned clone suspension from cultured hydrogel spheres and leave the sample in a small final volume of PBS for clone counting.

Constraints

1. Start with all hydrogel spheres together with their culturing ESC medium in a 1.5 mL tube.
2. Add dextranase at a 1:20 dilution and incubate at 37 C for 30 min so the PEGDT-MALDEX hydrogel dissolves.
3. After dissolution, gently mix the suspension before filtration.
4. Use a 70 um cell strainer to remove hydrogel remnants.
5. After filtration, let the clones sink by gravity at 37 C for 30 min before removing liquid.
6. The settled clone fraction is loose, so remove upper liquid by pipette handling rather than vacuum-style aspiration.
7. Replace removed medium with 1 mL of 1 x PBS, repeat the PBS wash once, and finish with the clone suspension in 200 uL of 1 x PBS.

Available Inputs

clone_culture_tube: 1.5 mL tube containing hydrogel-grown clonal spheres and ESC medium.
dextranase: dextranase reagent for PEGDT-MALDEX digestion.
pbs_1x: 1 x PBS used for gentle clone washing and final resuspension.
cell_strainer_70um: 70 um cell strainer for removing hydrogel debris.
counting_tube: clean destination tube for the filtered and washed clone suspension.

Action Pool

add_reagent *Adds a reagent to a sample with a specified volume and mixing mode.*
Inputs: sample, reagent, volume_ul, mixing
Output: updated sample handle

incubate_sample *Holds a sample under a defined temperature and condition for a fixed duration.*
Inputs: sample, temperature_c, duration_min, condition
Output: incubated sample handle

mix_sample *Combines or resuspends material using a specified method, time, and speed.*
Inputs: sample, method, duration_s, speed
Output: mixed sample handle

filter_sample *Passes a sample through a defined filter into a destination vessel.*
Inputs: sample, filter_type, pore_size_um, destination
Output: filtered sample handle

discard_supernatant *Removes the liquid phase above a settled fraction using a specified method.*
Inputs: sample, method, volume_ul, retain
Output: settled fraction handle

resuspend_pellet *Resuspends a loose pellet or settled fraction in a specified reagent and final volume.*
Inputs: sample, reagent, volume_ul, method
Output: resuspended sample handle

Gold Action Sequence

1. **add_reagent**
   Input: clone_culture_tube
   Parameters: reagent = dextranase; volume = 50 uL; mixing = gentle inversion
   Output: step1
2. **incubate_sample**
   Input: step1
   Parameters: 37 C; 30 min; condition = static
   Output: step2
3. **mix_sample**
   Input: step2
   Parameters: method = pipetting; duration = 5 s; speed = gentle
   Output: step3
4. **filter_sample**
   Input: step3
   Parameters: filter = cell_strainer_70um; pore size = 70 um; destination = counting_tube
   Output: step4
5. **incubate_sample**
   Input: step4
   Parameters: 37 C; 30 min; condition = static
   Output: step5
6. **discard_supernatant**
   Input: step5
   Parameters: method = pipetting; volume = 1000 uL; retain = true
   Output: step6
7. **add_reagent**
   Input: step6
   Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle pipetting
   Output: step7
8. **incubate_sample**
   Input: step7
   Parameters: 37 C; 30 min; condition = static
   Output: step8
9. **discard_supernatant**
   Input: step8
   Parameters: method = pipetting; volume = 1000 uL; retain = true
   Output: step9
10. **add_reagent**
    Input: step9
    Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle pipetting
    Output: step10

11. **incubate_sample**
    Input: step10
    Parameters: 37 C; 30 min; condition = static
    Output: step11
12. **discard_supernatant**
    Input: step11
    Parameters: method = pipetting; volume = 1000 uL; retain = true
    Output: step12
13. **resuspend_pellet**
    Input: step12
    Parameters: buffer = pbs_1x; volume = 200 uL; method = gentle pipetting
    Output: step13

Model Action Sequence: Claude Opus 4.7

1. **add_reagent**
    Input: clone_culture_tube
    Parameters: reagent = dextranase; volume = 50 uL; mixing = gentle pipetting
    Output: digested
2. **incubate_sample**
    Input: digested
    Parameters: 37 C; 30 min; condition = static
    Output: dissolved
3. **mix_sample**
    Input: dissolved
    Parameters: method = pipetting; duration = 10 s; speed = gentle
    Output: mixed
4. **filter_sample**
    Input: mixed
    Parameters: filter = cell_strainer_70um; pore size = 70 um; destination = counting_tube
    Output: filtered
5. **incubate_sample**
    Input: filtered
    Parameters: 37 C; 30 min; condition = static gravity sedimentation
    Output: settled
6. **discard_supernatant**
    Input: settled
    Parameters: method = pipetting; volume = 1000 uL; retain = true
    Output: after_removal_1
7. **add_reagent**
    Input: after_removal_1
    Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle pipetting
    Output: pbs_wash_1
8. **incubate_sample**
    Input: pbs_wash_1
    Parameters: 37 C; 30 min; condition = static gravity sedimentation
    Output: settled_1
9. **discard_supernatant**
    Input: settled_1
    Parameters: method = pipetting; volume = 1000 uL; retain = true
    Output: after_removal_2

10. **add_reagent**
    Input: after_removal_2
    Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle pipetting
    Output: pbs_wash_2
11. **incubate_sample**
    Input: pbs_wash_2
    Parameters: 37 C; 30 min; condition = static gravity sedimentation
    Output: settled_2
12. **discard_supernatant**
    Input: settled_2
    Parameters: method = pipetting; volume = 1000 uL; retain = true
    Output: after_removal_3
13. **resuspend_pellet**
    Input: after_removal_3
    Parameters: buffer = pbs_1x; volume = 200 uL; method = gentle pipetting
    Output: final_clone_suspension

Model Action Sequence: Qwen3.5-9B

1. **add_reagent**
   Input: clone_culture_tube
   Parameters: reagent = dextranase; volume = 75 uL; mixing = gentle_invert
   Output: digestion_mix
   Error: Dextranase volume differs from the gold 50 uL value and the mixing label differs from the action-pool style.
2. **incubate_sample**
   Input: digestion_mix
   Parameters: 37 C; 30 min; condition = static
   Output: digested_sample
3. **mix_sample**
   Input: digested_sample
   Parameters: method = inversion; duration = 60 s; speed = slow
   Output: mixed_sample
   Error: A pre-filtration mixing step exists, but its method and duration differ from the gold gentle 5 s pipetting step.
4. **filter_sample**
   Input: mixed_sample
   Parameters: filter = cell strainer; pore size = 70 um; destination = counting_tube
   Output: filtered_sample
5. **incubate_sample**
   Input: filtered_sample
   Parameters: 37 C; 30 min; condition = static
   Output: settled_clones
6. **discard_supernatant**
   Input: settled_clones
   Parameters: method = pipette; volume = 1000 uL; retain = true
   Output: pellet
7. **add_reagent**
   Input: pellet
   Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle_invert
   Output: first_wash

8. **discard_supernatant**
   Input: first_wash
   Parameters: method = pipette; volume = 1000 uL; retain = true
   Output: pellet_after_first
   Error: A settling incubation is missing before this removal step.
9. **add_reagent**
   Input: pellet_after_first
   Parameters: reagent = pbs_1x; volume = 1000 uL; mixing = gentle_invert
   Output: second_wash
10. **discard_supernatant**
   Input: second_wash
   Parameters: method = pipette; volume = 1000 uL; retain = true
   Output: final_pellet
   Error: A second settling incubation is also missing before this removal step.
11. **resuspend_pellet**
   Input: final_pellet
   Parameters: buffer = pbs_1x; volume = 200 uL; method = gentle_pipetting
   Output: final_clone_suspension

Evaluation

Claude Opus 4.7 Level 2 Action Sequence Similarity: 1.0000
Claude Opus 4.7 Level 2 Parameter Accuracy: 0.7692
Claude Opus 4.7 Level 2 Final Score: 0.8846
Qwen3.5-9B Level 2 Action Sequence Similarity: 0.0000
Qwen3.5-9B Level 2 Parameter Accuracy: 0.0000
Qwen3.5-9B Level 2 Final Score: 0.0000
Observed error pattern: The weaker baseline keeps the rough washing theme but changes the dextranase volume, changes the pre-filtration mixing parameters, and breaks two settling-removal dependencies.

D Evaluation Output Format

During evaluation, Level 1 predictions are parsed from the final next-action candidate after model reasoning. Level 2 predictions are parsed from the structured action-sequence output and scored with Python AST over action order, parameters, and variable dependencies. The AST parser is an evaluator implementation detail for extracting action names, parameters, outputs, and dependency structure from Level 2 action sequences.

E Ethics, Social Impact, and Intended Use

LabHorizon is intended to support academic research on laboratory 3D perception, protocol-aligned action prediction, and long-horizon experimental planning. Its expected social impact is positive: better evaluation and training data can help researchers identify failure modes before deploying AI systems in scientific workflows, improve reproducibility analysis, and support education around experimental planning.

The dataset does not contain personal privacy information. The visual component is based on laboratory assets and instrument views, while the planning component abstracts experimental procedures into action pools and action sequences. It does not include human identities, patient records, private user data, or personally identifying metadata.

We also design LabHorizon to avoid biological-risk escalation. The released tasks evaluate whether a model can select or organize laboratory actions under protocol-style constraints; they are not intended as autonomous wet lab execution instructions, biosafety guidance, clinical recommendations,

or assistance for harmful biological activity. Users should follow institutional biosafety rules, legal requirements, and domain-expert review when working with any real experimental protocol.

LabHorizon data, evaluation code, trained models, and agent designs should be used for academic research, benchmarking, model diagnosis, and education. They should not be used as the sole basis for high-stakes scientific, clinical, regulatory, industrial, safety-critical, or autonomous laboratory decisions. Any downstream deployment should require qualified human oversight and task-specific safety validation.