ForesightBench: Benchmarking Research Foresight in Large Language Models

Ruhaan Chopra

Independent Researcher
ruhaanchopra2005@gmail.com

Abstract

Mainstream autoresearch loops largely search by brute force, which works only while the search space is small. Beyond that threshold, an agent needs research taste: a sense of which experiments will pay off before running them. To assess frontier models on their ability to forecast empirical outcomes of given research setups, we introduce ForesightBench, built by an agentic pipeline that turns recent arXiv preprints into contamination-controlled benchmark samples, and evaluate fifteen frontier models on 71 samples from papers submitted after every published training cutoff. We reveal that while most models get the direction of the prediction right, they are consistently conservative in their predictions. Error is also highest on safety and interpretability papers (0.37), against 0.18 on vision. We then identify three failure modes of reasoning, and discover that faulty predictions often happen when models fail to take cognizance of the entire provided experimental setup and imprudently rely on their priors. Using this as our guiding principle, we introduce a lightweight inference-time intervention which decomposes the setup into threads and sub-threads. We test this harness built on top of pi against vanilla, out-of-the-box pi, tasking them both with creating a long-horizon agent. Across two seeds, the agent built by our custom harness outperforms the one built by vanilla pi on DABstep by 8 percentage points on average.

Introduction

Modern frontier LLMs are multi-trillion-parameter models (Fedus, Zoph, and Shazeer 2022; Kimi Team 2026) that hold vast swaths of knowledge absorbed in their parametric memory and maintain a deep record of human endeavour (Allen-Zhu and Li 2024; Calderon et al. 2026). Their capabilities are rapidly advancing in domains like coding (Kwa et al. 2025; Desai et al. 2026), and recent work has attempted to leverage these capabilities to advance scientific research (Lu et al. 2024; Novikov et al. 2025).

Specifically in the AI research domain, autoresearch is a growing paradigm that involves having agents come up with breakthroughs which outperform the baseline (Karpathy 2026; Kong et al. 2026). It started as a primitive setup, where an agent with access to compute iterated over a `train.py` file until a required metric (e.g., training loss below a threshold) was fulfilled, building on a line of ML-engineering agents and benchmarks (Wijk et al. 2024; Chan et al. 2025). Over the past year, autoresearch pipelines and detailed recursive self-improvement (RSI) loops have become commonplace (Weng et al. 2025; Ding et al. 2026).

But the core iteration still seems to be carried out by brute force, which becomes increasingly inefficient as the search space grows, at which point the process must rely on the agent's research taste to determine what to explore and what to ignore. Research taste is a nebulous concept and is notoriously hard to quantify. Therefore, we need the closest quantifiable proxy: a verifiable metric that is downstream of tasteful experimentation. We determine that to explore the search space fuzzily, an agentic system must have the capability to identify the specific research directions out of all available ones which will end up being the most fruitful. In other words, the underlying model must possess the reasoning ability required to predict the empirical outcomes of the research setup provided. This need for inculcating capabilities to forecast the results of future directions before committing has also been documented in previous work (Foster et al. 2026).

To benchmark the capability of large language models to predict empirical results of research setups, we introduce **ForesightBench**, built by an agentic pipeline that scrapes recent arXiv preprints and turns each controlled within-paper comparison into a sample. Each sample consists of the paper's introduction and methodology with all results removed, and the task is to predict the value each reported metric takes after the technique is applied.

We evaluate fifteen frontier models on a 71-sample held-out set (Table 5), and surprisingly, models from nine developers, open and closed alike, all land between 0.254 and 0.292 with overlapping 95% intervals. We reveal that on our scoring metric, the models often get the direction of change right (Table 6), but are consistently conservative and undershoot in their predictions (Table 4). We also notice that on our scoring metric, error is consistently high for safety and mechanistic interpretability preprints and among the lowest on vision research (Table 7). We show that faulty predictions often occur when a model fails to account for all moving parts of a given setup. Using this as our guiding principle, we design a lightweight agentic harness structured as a recursive fan-out tree, which decomposes the given brief into threads and sub-threads. This harness is built on top of pi (Zechner 2026), and to demonstrate the effect of this inference-level intervention, we task both it and vanilla pi with building

a long-horizon agent. The resulting agents are evaluated on the DABstep benchmark (Egg et al. 2025); averaged over two seeds, the agent built by our custom harness passes 46% of the held-out tasks against 38% for the agent built by vanilla pi, a gain of 8 percentage points (Table 10).

Benchmark Design and Statistics

Recent work has put LLMs in a forecasting position over research outcomes, but almost always through discrete choices. The model is asked which of two ideas will score higher on a benchmark (Wen et al. 2025; Mule, Garikaparthi, and Patwardhan 2026), which of two abstracts reports the real result (Luo et al. 2025), whether an advance will be realised or which of several candidate pathways or directions will pan out (Wu et al. 2026; Tian et al. 2026), or to pick from three or four options, with numeric items scored only as inside or outside an interval (Sehwag et al. 2026). These framings are brittle in a few ways: a choice between options implies that at least one is right, no baseline is given to anchor the answer, and a correct pick might not say anything about how large the effect was. Continuous targets in prior work take a different form from ours: PRECOG forecasts an absolute benchmark score from a masked description of the method (Park et al. 2025), and Zheng et al. compare agent-written code pairwise before execution (Zheng et al. 2026a).

We instead treat the LLM as an assistant to a researcher who has an exact methodology and experimental setup in mind, has already run the baseline on named benchmarks, and is mulling the idea over with the assistant rather than asking it to pick a winner. The prompt gives the full setup, the benchmarks, and the baseline numbers, and asks for the value each metric will take under the proposed method.

Because the answer is a number against a known baseline, every prediction implicitly carries a direction and a magnitude axis, which exposes behaviours a binary choice hides, such as conservatism toward the baseline or systematic over- or undershooting.

ForesightBench samples are built with our retrieval pipeline[1] (Appendix A; Figure 1 walks through one sample), which turns each controlled within-paper comparison in a recent arXiv preprint into a prompt with the results removed and a graded target. The papers come from a fixed arXiv submission window, which is the contamination control: for this corpus every paper was submitted between 8 and 24 August 2026 or between 5 and 21 October 2026, after the latest published training cutoff (June 2026) of every model whose vendor publishes one, and web search is disabled at inference. Three models (Grok 4.6, DeepSeek V4 Pro 0813, GLM-5.3) were released between 12 and 14 August without a published cutoff; we assume their training data realistically excludes papers submitted one to six days before release.

Every sample produced by the agentic pipeline was then audited by hand before entering the corpus, and we recommend the same manual audit to anyone using the pipeline to build their own foresight benchmark. The system prompt of every LLM stage, including the retrieval agent, is in the pipeline repository, and Appendix A names the file for each stage. A finished sample begins "I've been spending cycles on isolating how much" and ends with a question asking for the value of each (benchmark, metric) pair after switching from the baseline arm to the method arm; one complete sample with its target is reproduced verbatim in Appendix B.

[1] https://github.com/ruhzi/autoresearchrl

Table 1: Corpus and evaluation-set statistics. Samples per preprint are 1 in the evaluation set by construction.

	Corpus	Eval set
Preprints	71	71
Samples	457	71
Graded slots	3,343	581
Samples per preprint: mean (sd)	6.4 (6.2)	1
median	4	1
min / max	1 / 29	1 / 1
IQR outliers (preprints)	5	0
share of samples in top 3 preprints	17.1%	4.2%
Slots per sample: mean (sd)	7.3 (9.4)	8.2 (10.5)
median	5	4
min / max	1 / 68	1 / 68
Slots where the method value is higher	50.2%	47.0%
lower	45.6%	47.8%
equal to the baseline	4.2%	5.2%

Corpus and Evaluation Set

A single preprint can yield several distinct samples, one per controlled comparison it reports: each ablated component, each technique it introduces, and each setting it is tested under is treated as its own sample, because each different setting introduces a unique experimental context. The ForesightBench corpus consists of 457 samples from 71 preprints (Table 1); the corpus and the evaluation set are released together on GitHub[2] and Hugging Face.[3] Each sample asks for one or more graded slots (predictions), one per (benchmark, metric) pair, which means that the entire corpus contains 3,343 values to be predicted in all.

The median preprint contributes 4 samples; ten preprints contribute a single sample, and the most, 29, come from one preprint's ablation grid (arXiv 2608.09228; Ichihara et al. 2026). The evaluation set draws exactly one sample per preprint, chosen by a random seed, giving 71 samples with 581 slots to be predicted by the model under evaluation. The sign of the true change is close to balanced: the method value is higher than the baseline on 47% of evaluation slots, lower on 48%, and equal on 5% (Table 1), so a constant guess on the raw direction of change is right about half the time. Whether a higher value is better depends on the metric and is not tagged in the corpus, so this is a statement about the sign of the change, not about whether the method helped. We tag every preprint by hand with one of six research areas from its title and abstract (Table 2; definitions and per-preprint assignments, with a citation for every source preprint, in Appendix C).

[2] https://github.com/ruhzi/benchmark

[3] https://huggingface.co/datasets/ruhzi/foresightbench

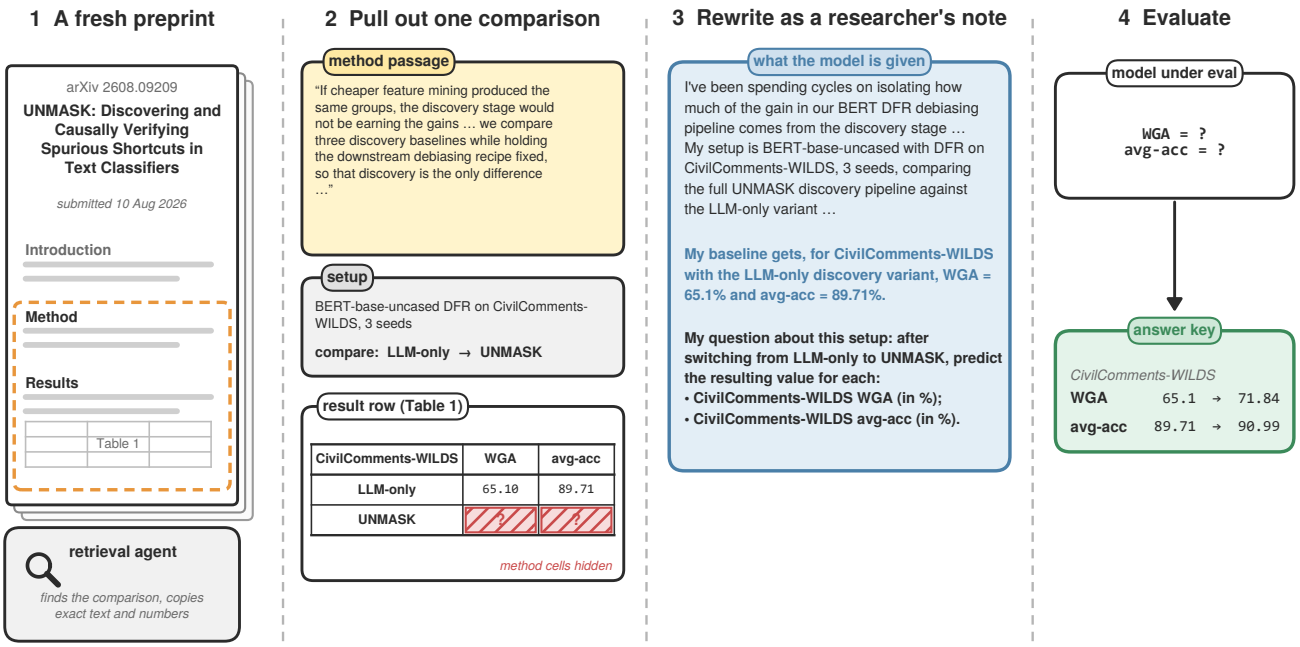

Figure 1: How one ForesightBench sample is built, shown on sample `2608.09209-s051` (Ravuru and Srivastava 2026). A retrieval agent pulls one controlled comparison out of a fresh preprint: the method passage, the experimental setup, and the result row, with the method cells hidden. The copied text and the baseline numbers are rewritten into a first-person researcher's note that ends with the question, and the hidden method cells become the answer key.

Table 2: Thematic coverage. Area is a per-preprint tag applied to all of its samples.

Area	Preprints	Corpus	Eval set
Vision & multimodal	16	135	16
Safety, attacks, interpretability	12	78	12
Agents, memory & context	15	56	15
Applied & scientific ML	17	108	17
LLM training & architecture	6	71	6
LLM evaluation studies	5	9	5

Evaluation

Models and Generation Settings

Fifteen frontier models (Table 3) were run on the ForesightBench evaluation set through OpenRouter and, for the six open-weight models on the items added in October, through Hugging Face Inference Providers, with web search disabled and one completion per sample. The longest completed answer was 48,243 tokens, by Gemini 4 Argon.

Scoring

We compute the error in the prediction as the relative distance between predicted and actual value, capped at one:

$$e = \min\left(\frac{|\hat{v} - v|}{\max(|v|, \epsilon)},\ 1\right), \qquad \epsilon = 10^{-6}, \tag{1}$$

and in case of an unexplained non-response, the error is recorded as 1.

Non-responses caused by the serving path (provider content filters, transport and provider errors, empty output with leaked control tokens, empty output at the token budget) are excluded for that (model, sample) pair only; slots the model itself leaves without a number are charged the cap of 1. Provider content filters account for every exclusion on the Claude models, so Claude Opus 5.5 is scored on 69 samples, Claude Fable 5.1 on 67 and Claude Fable 5 on 66. Gemini 4 Argon is scored on 68 and Kimi K3 on 70, both with samples lost on the serving path. Every other model is scored on all 71 (the n column of Table 3). Mean completion tokens per scored sample are reported beside every score; the count is the provider's billed completion total, which includes hidden reasoning tokens for every model.

As a reference, Table 5 contains a *copy-the-baseline* row (mean error if predicted and actual values were the same).

Because every slot also carries its baseline value b, a prediction decomposes into a direction and the boldness of the prediction, and we report both beside the error. The *direction* of a slot is right when $\hat{v} - b$ and $v - b$ have the same sign; it is evaluated on slots where the reported value moved ($v \neq b$), and a prediction that stays on the baseline counts as neither right nor wrong. The *move ratio* of a slot is the size of the predicted move relative to the reported one:

$$r = \frac{|\hat{v} - b|}{|v - b|}, \tag{2}$$

so $r = 1$ is a move of exactly the reported size, $r < 1$ an undershoot and $r > 1$ an overshoot. The ratio is computed only on slots whose direction is right.

Table 3: Models and generation settings. Every model was sampled at temperature 0.4 with reasoning at high effort and an output budget of 65,536 completion tokens. Via: serving route (HF: Hugging Face Inference Providers). n: samples scored.

Model	Via	n
GPT-6 Astra (OpenAI 2026b)	OpenRouter	71
Claude Opus 5.5 (Anthropic 2026c)	OpenRouter	69
Kimi K3 (Kimi Team 2026)	OpenRouter / HF	70
Claude Fable 5.1 (Anthropic 2026b)	OpenRouter	67
Claude Fable 5 (Anthropic 2026a)	OpenRouter	66
Gemini 4 Argon (Google DeepMind 2026)	OpenRouter	68
GPT-5.6 Sol (OpenAI 2026a)	OpenRouter	71
GPT-6 Sol (OpenAI 2026c)	OpenRouter	71
DeepSeek V4 Flash (DeepSeek-AI 2026)	OpenRouter / HF	71
GLM-5.3 (Z.ai 2026b; GLM-5-Team 2026)	OpenRouter / HF	71
GPT-6 Luna (OpenAI 2026c)	OpenRouter	71
Grok 4.6 (SpaceXAI 2026)	OpenRouter	71
Inkling (Thinking Machines Lab 2026)	OpenRouter / HF	71
DeepSeek V4 Pro (DeepSeek-AI 2026)	OpenRouter / HF	71
Beam 1.0 (Reflection AI 2026)	OpenRouter / HF	71

Table 4: Size of the predicted move off the baseline relative to the reported move (move ratio r, defined under Scoring), over the slots whose direction is right. Undershoot: share of those slots with $r < 1$. Overshoot: share with $r > 1$. Median ratio: median of r over the same slots.

Model	Undershoot	Overshoot	Median ratio
DeepSeek V4 Pro	**45%**	53%	1.11
GPT-6 Astra	52%	46%	0.94
GPT-5.6 Sol	54%	43%	0.88
DeepSeek V4 Flash	55%	42%	0.83
Claude Opus 5.5	61%	35%	0.69
GPT-6 Sol	61%	38%	0.75
Claude Fable 5	64%	36%	0.71
Claude Fable 5.1	65%	32%	0.64
Kimi K3	66%	33%	0.58
Gemini 4 Argon	66%	32%	0.70
Grok 4.6	66%	33%	0.64
Beam 1.0	66%	32%	0.64
GLM-5.3	67%	31%	0.60
Inkling	67%	32%	0.55
GPT-6 Luna	70%	28%	0.41

Results

The results are revealing, and a striking behavioural trait is conservatism. Fourteen of the fifteen models undershoot more often than they overshoot (Table 4): on the slots where the direction is right, the predicted move is smaller than the reported one on 45% to 70% of slots; DeepSeek V4 Pro, the exception, overshoots slightly more often than it undershoots and its median move is 1.11 of the reported one. The models tend to be largely directionally correct (72% to 85% of slots, Table 6) but conservative in their final empirical predictions.

Mean per-sample error on the 71-sample evaluation set lies between 0.254 and 0.292 for all fifteen models (Table 5; medians by area in Table 15, Appendix D). Copying the baseline, which is to say assuming the intervention did nothing, has a mean error of 0.327. The 95% intervals of all fifteen models overlap: surprisingly, despite their differences in vendor, scale and reasoning budget, the models fall around one another on error.

GPT-6 Astra is the modest outperformer. It has the lowest mean error (0.254; Table 5), it is among the boldest models in the field (undershoot on 52% of its direction-right slots, a median move of 0.94 of the reported one, second only to DeepSeek V4 Pro; Table 4), and it has the best directional accuracy (85%, against 82% for GPT-6 Sol and Gemini 4 Argon; Table 6). Yet even Astra undershoots more often than not. At the other end, GPT-6 Luna is the most timid: it undershoots on 70% of its direction-right slots and its median move is 0.41 of the reported one (Table 4).

Spending more reasoning tokens does not improve the result. Completion tokens per sample range from 1,104 for Claude Fable 5 to 13,050 for DeepSeek V4 Pro (Table 5; generation settings in Table 3), more than a tenfold spread in reasoning, and the extra reasoning yields no significant gains. The best model in the field, Astra, is also one of the most frugal at 1,754 tokens. Thinking longer does not make a model less conservative.

GPT-6 Astra predicts the direction of the intervention correctly most often, on 85% of slots, GPT-6 Sol and Gemini 4 Argon on 82%, and the rest of the frontier on 72% to 81% (Table 6).

There is no correlation between the number of slots a sample asks for and the error on that sample, which suggests that the error is not a function of the number of predictions the model must make (Table 8; Figure 2).

Notably, pooled over all models, error is highest on safety and interpretability papers (0.37), applied ML (0.32) and agent papers (0.29), and lowest on vision (0.18) and LLM evaluation studies (0.14) (Table 8). The split holds for every model individually (Table 7): each model's error on vision and LLM evaluation papers is below its error on safety and agent papers, although the order within each pair varies from model to model (area definitions in Appendix C).

After conducting a qualitative reading of every scored trace, we identified three failure modes of reasoning. Not every provider returns the chain of thought itself: the Anthropic, OpenAI, xAI and Google models return a provider-written summary of the reasoning (the stored trace runs 0.4 to 1.4 characters per billed reasoning token, against roughly four for the models that return the full trace), so for those models the reading is of the summary and the visible answer. Every scored trace of the fifteen models, with its research-area label and failure-mode tags, is released alongside the evaluation set in the benchmark repository.

Mode 1, *unfaithful chain of thought*: the committed conclusion is not a function of the reasoning that precedes it. On sample `2608.17941-s010` (Liu et al. 2026d), Kimi K3 reasons that the predicted value should fall below the baseline value, writing "that argues for degradation, particularly in correlation", yet perplexingly predicts a value above the baseline value; the actual value did fall. This mode is well

Table 5: Error by model. Mean: capped relative error averaged over the evaluation samples, with its 95% paired percentile-bootstrap interval over samples ($B = 4000$; every model is scored on the same resamples). Median: median per-sample error. Tok./sample: mean completion tokens.

Model	Mean	95% CI	Median	Tok./sample
GPT-6 Astra	**0.254**	[0.189, 0.324]	0.128	1,754
Claude Opus 5.5	0.256	[0.190, 0.327]	0.126	1,224
Kimi K3	0.259	[0.194, 0.330]	0.138	5,906
Claude Fable 5.1	0.262	[0.194, 0.338]	0.128	1,328
Claude Fable 5	0.265	[0.198, 0.339]	0.156	1,104
Gemini 4 Argon	0.266	[0.195, 0.341]	0.153	11,049
GPT-5.6 Sol	0.267	[0.200, 0.340]	0.148	3,014
GPT-6 Sol	0.270	[0.205, 0.343]	0.145	1,993
DeepSeek V4 Flash	0.272	[0.207, 0.344]	0.136	12,489
GLM-5.3	0.273	[0.206, 0.346]	0.145	2,601
GPT-6 Luna	0.277	[0.209, 0.352]	0.145	2,253
Grok 4.6	0.277	[0.208, 0.351]	**0.120**	5,195
Inkling	0.282	[0.212, 0.358]	0.147	3,624
DeepSeek V4 Pro	0.288	[0.219, 0.364]	0.150	13,050
Beam 1.0	0.292	[0.220, 0.369]	0.174	4,757
Copy baseline	0.327	[0.252, 0.404]	0.198	–

documented in recent work on chain-of-thought faithfulness (Chen et al. 2025; Jia, Benton, and Easley 2026; Gema et al. 2026).

Mode 2, *lapse in cognizance of the setup*: the trace fails to take the entire stated setup into account, often relying on priors that do not apply in that context, and the reasoning derails from there. On sample `2608.10777-s004` (Liao et al. 2026), the prompt states that the method's sampled trajectories run only over a short lookahead window and replace the baseline's full-horizon integration; Beam 1.0 fails to take cognizance of this and relies on a generic prior it holds, "PI-VM uses N sample trajectories, so slower", predicting a runtime above the baseline value; the actual value fell well below the baseline value.

Mode 3, *proposal bias*: beliefs about what the authors or the asker want influence the conclusion, in either direction. On sample `2608.10989-s023` (Cao et al. 2026a), Inkling considers that the attention-mean criterion may be no better than random, dismisses it with "why would they ask then?", and commits a smaller accuracy drop than random on all three tasks; the true drop was larger on two of them. This mode is well documented in recent work on sycophancy (Sharma et al. 2024; Feng et al. 2026; Nogueira et al. 2026).

Traces carrying mode 2 average an error of 0.51 against 0.23 for traces without it, a gap that holds for every model (Table 9; all three modes in Table 16, Appendix D).

On 32 of the 71 samples all models get the direction right, while on TRACE-Memory (`2608.08446-s005` (Wang et al. 2026c)) and attention-mean pruning (`2608.10989-s023` (Cao et al. 2026a)) every model gets the direction wrong on most slots. BIRD-RL (`2610.29426-s018` (Qu et al. 2026)), BatchBench (`2610.30104-s000` (Topf, Sanchez-Fernandez, and Klambauer 2026)) and SSE-Bio (`2608.22132-s041` (Meng et al. 2026)) are also challenging, with nine or more of the fifteen models getting the direction wrong on most slots. Two causes recur. Models assume the proposal of the sample helps, so when it hurts (TRACE-Memory, attention-mean pruning, OPSD padding in `2608.09228-s001` (Ichihara et al. 2026)) nearly all of them predict a gain, a case of proposal bias (Mode 3). They also assume cheaper means worse, a misplaced prior of Mode 2 that fails in ACM+TA merging (`2608.10447-s016` (Le et al. 2026); cheaper and better) and StagedWorkspace (`2608.18050-s006` (Hua et al. 2026); costlier and better, predicted cheaper and worse).

Table 6: Direction of predicted change.

Model	Direction right
GPT-6 Astra	**85%**
Gemini 4 Argon	82%
GPT-6 Sol	82%
Claude Fable 5.1	81%
Claude Fable 5	81%
GPT-5.6 Sol	79%
DeepSeek V4 Flash	78%
Kimi K3	77%
Inkling	76%
Claude Opus 5.5	75%
GLM-5.3	75%
DeepSeek V4 Pro	74%
Grok 4.6	73%
Beam 1.0	73%
GPT-6 Luna	72%

Autoresearch-Harness

Dissecting the reasoning traces of these models and understanding where they fail at forecasting research outcomes lead us to theorise about the underlying principles we can derive from them. We derive the following driving principle, which follows from our analysis of failure modes and from our initial hypothesis that focusing on pre-experimentation forecasting would yield better autoresearch than current loops: the autoresearch loop must take cognizance of the entire setup before beginning implementation.

There are two ways to test this; the obvious one is a post-training environment designed around this principle. Budgetary constraints rule this out, so we instead test whether an inference-level intervention can have a sizeable effect. To that end, we design a lightweight agentic harness built on top of pi, and task it with a small autoresearch brief.

The harnesses (or autoresearchers) are tasked with creating a long-horizon agent, with a provided brief on the sort of tasks that agent must carry out (reproduced in full in Appendix E.1). We then evaluate the agents built by these autoresearchers on a set of held-out tasks, and compare the performance of the agents built by the harnesses.

The custom harness

The core tenet of the custom harness is that the researcher takes cognizance of the entire setup before it implements

Table 7: Error by research area.

Model Samples	Vision 16	Safety 12	Agents 15	Applied 17	LLM-train 6	LLM-eval 5	All 71
GPT-6 Astra	0.17	0.33	0.26	0.31	0.24	0.13	**0.25**
Claude Opus 5.5	0.19	**0.31**	0.29	0.31	**0.21**	**0.12**	0.26
Kimi K3	0.18	0.38	0.27	**0.27**	0.24	0.16	0.26
Claude Fable 5.1	0.18	0.39	0.28	0.30	0.25	0.14	0.26
Claude Fable 5	0.20	0.33	0.29	0.33	0.24	0.14	0.27
Gemini 4 Argon	**0.16**	0.35	0.30	0.29	0.39	**0.12**	0.27
GPT-5.6 Sol	**0.16**	0.37	0.29	0.33	0.22	0.15	0.27
GPT-6 Sol	0.19	0.37	**0.25**	0.34	0.24	0.15	0.27
DeepSeek V4 Flash	0.17	0.40	0.32	0.29	0.24	**0.12**	0.27
GLM-5.3	0.20	0.39	0.30	0.30	0.25	**0.12**	0.27
GPT-6 Luna	0.20	0.43	0.26	0.30	0.25	0.16	0.28
Grok 4.6	0.20	0.39	0.26	0.34	0.25	0.13	0.28
Inkling	0.19	0.37	0.32	0.33	0.25	0.15	0.28
DeepSeek V4 Pro	0.17	0.37	0.33	0.36	0.25	0.14	0.29
Beam 1.0	0.21	0.35	0.33	0.35	0.25	0.13	0.29

Table 8: Error and slots per area.

Research area	Samples	Slots/sample	Mean error
Vision	16	5.2	0.18
Safety & interpretability	12	5.2	0.37
Agents	15	6.5	0.29
Applied ML	17	12.3	0.32
LLM training & architecture	6	8.5	0.25
LLM evaluation studies	5	15.2	0.14
All	71	8.2	0.27

Table 9: Mode 2 by model.

Model	Traces	Mode 2	With	Without
Claude Opus 5.5	69	**5**	0.58	0.23
GPT-6 Astra	71	7	0.46	0.23
Claude Fable 5	66	7	0.48	0.24
GPT-6 Sol	71	7	0.44	0.25
Claude Fable 5.1	67	8	0.57	0.22
GPT-6 Luna	71	9	0.49	0.25
Gemini 4 Argon	68	11	0.42	0.24
GPT-5.6 Sol	71	11	0.43	0.24
Kimi K3	70	12	0.56	0.20
DeepSeek V4 Flash	71	13	0.50	0.22
DeepSeek V4 Pro	70	13	0.44	0.25
GLM-5.3	71	14	0.55	0.21
Grok 4.6	71	14	0.56	0.21
Beam 1.0	71	14	0.54	0.23
Inkling	70	16	0.57	0.20
All	1048	161	0.51	0.23

anything. The harness is built on top of vanilla pi (version 1.1.0).[4]

A root pass reads the task (the brief) and deduces its critical *threads*: the parts of the setup that the build depends on. Each thread must quote the brief's own words and state why it is crucial. One further pass per thread then breaks it into sub-threads in the same way. The depth of this recursive fan-out tree is configurable; we chose a depth of two (threads and sub-threads) and a cap of six threads per pass.

The root conversation is then given one more turn in which it decides the chronology: the order in which the threads are to be built, by dependency. Implementation follows that order. Each first-level thread's conversation is continued into an implementation turn that builds its sub-threads in a git worktree of the workspace, one thread at a time, and every implementation turn receives the reports of the turns before it and may change earlier work. The combined changes are held as a diff outside the workspace.

Meanwhile, a researcher pass receives the brief and is asked to reason about the implementation. It is then given the implementers' diff, with the thread tree and each implementer's report, and asked to reconcile its own plan with them. From there the run is an ordinary pi session until the budget is spent.

[4]https://github.com/ruhzi/autoresearch-harness

Setup

Both arms run inside the same sandboxed arena. The outer (researcher) model is GLM 5.3 Flash (Z.ai 2026a; GLM-5-Team 2026) in both arms. The inner model, which the built agent calls and which the researcher may spend on practice runs, is GPT-6 Luna (OpenAI 2026c); the inner budget is $4.00 of calls per run, and the run ends when it is spent or the researcher declares itself done. The brief describes the genre of task only (long-horizon quantitative questions over a workspace of files and their documentation).

The agents are evaluated on DABstep, a benchmark of multi-step data-analysis questions over a shared set of files and documentation, graded against the reference answer with a numeric tolerance. We draw a held-out set of 75 tasks from its *hard* level. During evaluation each agent runs one fresh process per task on GPT-6 Luna with no network and no memory across tasks.

Each autoresearcher is run twice (seeds 1 and 2). A seed is

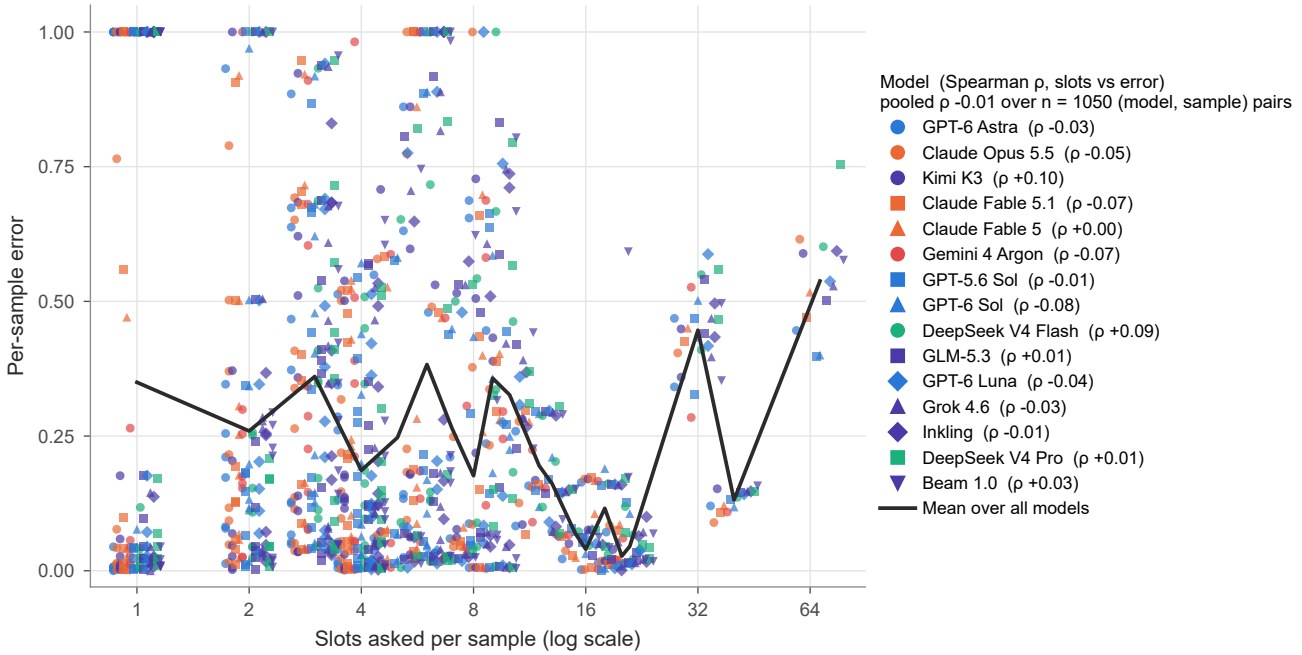

Figure 2: Per-sample error against the number of slots the sample asks for, every model in one panel. One point per (model, sample) pair; hue is the model family and marker the model, with a small horizontal offset per model so points at the same slot count do not overlap. The black line is the mean error over all models at each slot count. Spearman ρ between slots and error is given per model in the legend.

Table 10: Agents built by each autoresearcher (the mean inner-model cost of one evaluation task in US cents and the median wall-clock seconds per task.

	Pass rate (%)			¢ per task		Median s	
Autoresearcher	S1	S2	Mean	S1	S2	S1	S2
Vanilla pi	40.0	36.0	38.0	0.86	0.80	107	120
Custom	**44.0**	**48.0**	**46.0**	**0.57**	**0.43**	**55**	**50**

a complete run of the outer autoresearcher, so each arm yields two independently built agents, and every agent is scored on all 75 tasks.

Results

Table 10 reports the pass rate of each built agent. The vanilla pi agents pass 40.0% and 36.0% of the 75 tasks, a mean of 38.0%. The custom-harness agents pass 44.0% and 48.0%, a mean of 46.0%. The custom agents are also cheaper and faster at evaluation time: 0.57 and 0.43 cents of inner-model spend per task against 0.86 and 0.80 for the vanilla agents, and a median of 55 and 50 seconds per task against 107 and 120. Building the agents spent almost none of the $4.00 inner budget (Table 11); no run reached the cap, and all four ended with the researcher declaring itself done. With two seeds per arm the comparison is indicative rather than conclusive. The harness, the brief and the four agents are released for reproduction.

Table 11: Inner-model spend (USD) of each autoresearcher run while building its agent, against a budget of $4.00 per run.

Autoresearcher	S1	S2
Vanilla pi	0.04	0.02
Custom	0.05	0.03

Limitations

The failure modes were identified from a close reading of every trace rather than through controlled annotation, and some traces are provider summaries. We consider the reading a faithful account of what the traces do, and controlled annotation of the same traces is the natural next step to confirm the counts. We have released the tagged traces for external auditing.

The answer keys are the numbers the preprints report; ForesightBench takes correct reportage as given, by design.

Three models (Grok 4.6, DeepSeek V4 Pro 0813, GLM-5.3) publish no training cutoff, so their contamination control rests on our assumption that their training data excludes papers submitted one to six days before their release.

The autoresearch-harness comparison is small: two harnesses, two seeds each, one brief, and one benchmark. At this scale we view it as a demonstration and not a definitive evaluation. Larger autoresearch harnesses, more seeds, and more demanding evaluations are left to future work.

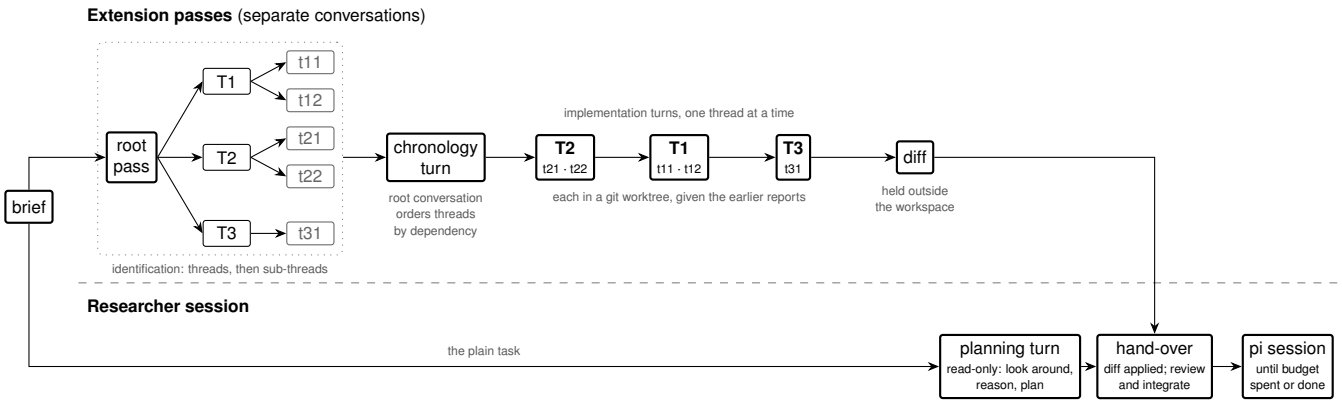

Figure 3: The custom harness. A root pass reads the brief and deduces its critical threads (T), and one further pass per thread deduces its sub-threads (t); depth two, at most six threads per pass. The root conversation then orders the threads by dependency, and each thread's conversation is continued into an implementation turn that builds its sub-threads in that order. The combined diff is held outside the workspace while the researcher plans with edits refused, and is applied when the researcher replies without tool calls; the researcher then reviews and integrates the build, and the run continues as an ordinary pi session. Every prompt is in the released harness source.

Since the retrieval pipeline is configurable and open-source, future work can expand this style of benchmarking into other scientific domains.

Acknowledgements

AI assistants helped with code, writing and trace annotation. We reviewed all output and take full responsibility.

References

Allen-Zhu, Z.; and Li, Y. 2024. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws. *arXiv preprint arXiv:2404.05405*.

Almeida, R. L.; and Pöhner, I. 2026. Fragment-Level Graph Encoding for Interpretable Molecular Property Prediction. *arXiv preprint arXiv:2610.29589*.

Alva, J.; Harper, D.; Thornton, A. E.; Alva, R. S. V.; Suazo, L. K. M.; Karageorgos, K.; Velazquez, D.; Serrano, S. A.; Flores, M. V.; Schliem, A.; and Migone, F. F. 2026. Global Security Blind Spots: Translation as a Safety Bypass in Low-Resource Languages. *arXiv preprint arXiv:2610.27304*.

Anthropic. 2026a. Claude Fable 5 and Mythos 5 System Card. https://www.anthropic.com/claude-fable-5-mythos-5-system-card. June 2026.

Anthropic. 2026b. Claude Fable 5.1 and Mythos 5.1 System Card. https://www.anthropic.com/claude-fable-5-1-mythos-5-1-system-card. September 2026.

Anthropic. 2026c. Claude Opus 5.5 System Card. https://www.anthropic.com/claude-opus-5-5-system-card. September 2026.

Bang, Y.; Fielding, K.; Oliver, B.; Birke, B.; Seedat, N.; and Bean, A. M. 2026. ContractScrub: A benchmark for final review of legal contracts. *arXiv preprint arXiv:2608.20204*.

Bhadane, J.; Yerramsetti, R.; Gupta, R.; and Kumar, S. 2026. Runtime Control for AI Agents: Closing the Evaluation–Execution Gap in Operational Workflows. *arXiv preprint arXiv:2610.27480*.

Blasingame, Z. W.; Franca, A.; Minan, R.; Koke, C.; Rindtorff, N.; and Tong, A. 2026. Improved Training of Strong Stochastic Flow Maps. *arXiv preprint arXiv:2610.29605*.

Cai, Z.; Ziakas, C.; Kontonis, V.; Pearce, T.; Sen, S.; Krishnamurthy, A.; Garg, S.; and Papailiopoulos, D. 2026. Fork-and-Flush: Escaping Idea Basins in Autoresearch Agents. *arXiv preprint arXiv:2610.07447*.

Calderon, N.; Ben-David, E.; Gekhman, Z.; Ofek, E.; and Yona, G. 2026. Empty Shelves or Lost Keys? Recall Is the Bottleneck for Parametric Factuality. *arXiv preprint arXiv:2602.14080*.

Cao, H.; Jaber, M.; Yuan, S.; and Sayed, A. 2026a. Putting Registers to Work: Task Registers for Token Pruning in Vision Transformers. *arXiv preprint arXiv:2608.10989*.

Cao, J.; Li, Q.; Zhang, Z.; Wu, X.; Liu, L.; Li, X.; and Ni, J. 2026b. MarkNull: Model-Agnostic Watermark Removal in AI-Generated Images via On-Manifold Latent Manipulation. *arXiv preprint arXiv:2608.10166*.

Cao, Y.; Wang, G.; Hou, M.; Bu, X.; Liu, D.; and Liu, Y. 2026c. EgoHieraLoc: A Cortically Inspired Hierarchical Segmentation-Guided Framework for Egocentric Visual Query Localization. *arXiv preprint arXiv:2608.09656*.

Chan, J. S.; Chowdhury, N.; Jaffe, O.; Aung, J.; Sherburn, D.; Mays, E.; Starace, G.; Liu, K.; Maksin, L.; Patwardhan, T.; Weng, L.; and Mądry, A. 2025. MLE-bench: Evaluating Machine Learning Agents on Machine Learning Engineering. In *International Conference on Learning Representations (ICLR)*.

Chen, B.; Zi, H.; Ou, R.; Liu, Y.; Chen, M.; Qi, Z.; Wang, R.; and Liu, G. 2026a. Pragmatic Attack Surface: Vulnerabili-

ties of Implicit Context in Large Language Models. *arXiv preprint arXiv:2608.09551*.

Chen, G.; Li, Z.; Li, R.; Lu, Y.; Shi, H.; Xu, B.; and Huang, J. 2026b. ForestBench: A Unified Graph Framework for Evaluating Multi-Agent Collaboration. *arXiv preprint arXiv:2608.08605*.

Chen, Y.; Benton, J.; Radhakrishnan, A.; Uesato, J.; Denison, C.; Schulman, J.; Somani, A.; Hase, P.; Wagner, M.; Roger, F.; Mikulik, V.; Bowman, S. R.; Leike, J.; Kaplan, J.; and Perez, E. 2025. Reasoning Models Don't Always Say What They Think. *arXiv preprint arXiv:2505.05410*.

Chlon, L.; and Huang, J. 2026. When Thoughts Become Memory: Unifying In-Context Learning and Fine-Tuning. *arXiv preprint arXiv:2610.29419*.

D'Alessandro, A.; Mahdavi-Amiri, A.; and Hamarneh, G. 2026. RECOUNT: Reference-guided Counting with Synthetic Visual Exemplars. *arXiv preprint arXiv:2608.20621*.

DeepSeek-AI. 2026. DeepSeek-V4: Towards Highly Efficient Million-Token Context Intelligence. arXiv preprint arXiv:2606.19348.

Deng, J.; Wu, X.; Wu, Z.; Wang, S.; Wang, Y.; and Zhu, C. 2026. WorldSR: Harnessing World Knowledge Search for Grounded Image Super-Resolution. *arXiv preprint arXiv:2610.28423*.

Desai, R.; Hu, J.; Cabezas, J.; Harsola, N.; Shukla, P.; Ben Chaim, R.; El Assadi, A.; Kamath, O. M.; Faldu, F.; Hebbar, P.; et al. 2026. SWE-Marathon: Can Agents Autonomously Complete Ultra-Long-Horizon Software Work? *arXiv preprint arXiv:2606.07682*.

Ding, T.; Nannapaneni, A.; Liu, B.; and Zhang, L. 2026. Autonomous Research Agents: A Survey of AI Scientists and the Verification Gap. *arXiv preprint arXiv:2608.05179*.

Du, Y.; Zhang, Y.; and Li, M. 2026. Hallucinations of Large Language Models in Cross-Language Code Translation: Phenomena and Mitigation. *arXiv preprint arXiv:2610.27868*.

Egg, A.; Iglesias Goyanes, M.; Kingma, F.; Mora, A.; von Werra, L.; and Wolf, T. 2025. DABstep: Data Agent Benchmark for Multi-step Reasoning. *arXiv preprint arXiv:2506.23719*.

Elhagrasy, Y. A.; Hill, I.; and Ivanov, A. 2026. Adaptive Sequential Test Planning for Multi-Mechanism Reliability Qualification via Bayesian Monte Carlo Tree Search. *arXiv preprint arXiv:2608.09622*.

Erramilli, V. 2026. Mechanistic Tomography: Designed Measurement for Control-Oriented Interpretability. *arXiv preprint arXiv:2608.19338*.

Fang, G.; Qian, S.; Lu, D.; Huang, J.; Jiang, L.; Wang, X.; Song, Y.; Jiang, L.; and Shou, M. Z. 2026. SolarRenderer: Few-Step Generative Reference Rendering. *arXiv preprint arXiv:2610.28630*.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23(120): 1–39.

Feng, Z.; Chen, Z.; Ma, J.; Po, Y. T.; Chersoni, E.; and Li, B. 2026. Good Arguments Against the People Pleasers: How Reasoning Mitigates (Yet Masks) LLM Sycophancy. *arXiv preprint arXiv:2603.16643*.

Foster, T. S.; Al Omari, B.; Fu, T.; Mann, T.; Domond, C.; Cipolina-Kun, L.; Gauri, B.; Aghamelu, M.; Goldie, A. D.; Helenowski, E.; Gagnon-Audet, J.-C.; Pepe, A.; Nazir, S.; Izcovich, D.; Levi, N.; Hazra, R.; Hambardzumyan, K.; Baldwin, N.; Li, X.; Josifoski, M.; Giampouras, P.; Jalili Sabet, M.; Sims, A.; Momand, H.; Shavrina, T.; Magka, D.; Weston, J.; Wang, Y.; Goyal, A.; Henriques, J.; Bachrach, Y.; McMilin, E.; and Foerster, J. N. 2026. AI Research Preference Models. *arXiv preprint arXiv:2608.13940*.

Gema, A. P.; Rajani, N.; Saxena, R.; Kwan, W.-C.; and Minervini, P. 2026. Chain-of-Thought Faithfulness of Reasoning Models Varies with Where and How Preference Cues Are Delivered. *arXiv preprint arXiv:2608.29464*.

GLM-5-Team. 2026. GLM-5: from Vibe Coding to Agentic Engineering. arXiv preprint arXiv:2602.15763.

Godey, N.; and Artzi, Y. 2026. Matryoshka Language Model Suites. *arXiv preprint arXiv:2608.09703*.

Google DeepMind. 2026. Gemini 4 Argon Model Card. https://deepmind.google/models/model-cards/gemini-4-argon/. October 13, 2026.

Grashei, C.; Gülhan, F.; Legnar, M.; Stögbauer, F.; Weis, C.-A.; Mogler, C.; and Schüffler, P. 2026. An end-to-end-trained vision-language model for native-language prostate pathology report generation. *arXiv preprint arXiv:2608.23143*.

Gálvez, R. H.; Venturini, G. M.; and Mondrzak, I. 2026. Coincidence-Maximizing Customer Clustering Using Smart Meter Data. *arXiv preprint arXiv:2610.28178*.

Holzwarth, L.; González-Márquez, R.; and Kobak, D. 2026. Most biomedical publications show signs of LLM-assisted writing. *arXiv preprint arXiv:2608.10715*.

Hua, Y.; Na, H.; Zhou, Y.; Kalose, A.; Ayubcha, C.; and Lian, L. 2026. StagedWorkspace: A Versioned Workspace for Knowledge-Work Agents. *arXiv preprint arXiv:2608.18050*.

Ichihara, Y.; Iwase, N.; Quamar, M. A.; and Komiyama, J. 2026. Privileged Solutions or Context-Induced Teacher Behavior? Dissecting On-Policy Self-Distillation. *arXiv preprint arXiv:2608.09228*.

Jia, J.; Benton, J.; and Easley, E. 2026. Faithfulness as Information Flow: Evaluating and Training Faithful Chain-of-Thought Reasoning. *arXiv preprint arXiv:2605.24286*.

Jiang, X.; Xie, R.; and Tang, M. 2026. SkillSmith: Enhancing Locally Deployed Agents via Automatic Skill Construction and Evolution. *arXiv preprint arXiv:2608.08037*.

Jin, S.; Kim, K.; Lee, J.; and Lee, Y. 2026a. Retrieval-Corrected Conformal Prediction for Time Series. *arXiv preprint arXiv:2608.10553*.

Jin, T.; Dong, S.; Li, H.; and Chou, C.-H. 2026b. Iterative Erasure Count Is Not an Affine-Invariant Concept Dimension. *arXiv preprint arXiv:2608.10566*.

Jo, S.; Ko, D.; Kang, J.; Kwak, Y.; Kwak, J.; and Choi, S. 2026. Flow Straight to Reality: Perceptually Consistent Flow Matching for Efficient Image Restoration. *arXiv preprint arXiv:2608.10544*.

Johnson, A.; Ghizawi, O.; and Minai, A. A. 2026. The Role of Grid Cells in Reducing Spatial Aliasing in Hippocampal Place Representations. *arXiv preprint arXiv:2608.18569*.

Kampe, J. N.; Lee, C. J.; Shen, X.; Lehtiö, A.; von Brandenburg, S.; Nokelainen, O.; Dunson, D. B.; and Ovaskainen, O. 2026. Physics-Informed Learning for Robust Acoustic Localization with Calibrated Uncertainty. *arXiv preprint arXiv:2608.08911*.

Karpathy, A. 2026. autoresearch: Autonomous LLM Research on a Single GPU. https://github.com/karpathy/autoresearch. GitHub repository, March 2026.

Kimi Team. 2026. Kimi K3: Open Frontier Intelligence. *arXiv preprint arXiv:2607.24653*.

Ko, J.-C. 2026. Uncensored Is Not Unbiased: Abliteration Removes Refusals but Not Script-Dependent Bias About Taiwan. *arXiv preprint arXiv:2610.29342*.

Kong, L.; Sun, X.; Chow, W.; Li, L.; Lin, K. Q.; Zhang, X. B.; Wang, S.; Li, R.; Wu, Q.; Gao, W.; et al. 2026. AI for Auto-Research: Roadmap & User Guide. *arXiv preprint arXiv:2605.18661*.

Kwa, T.; West, B.; Becker, J.; Deng, A.; Garcia, K.; Hasin, M.; Jawhar, S.; Kinniment, M.; Rush, N.; Von Arx, S.; et al. 2025. Measuring AI Ability to Complete Long Software Tasks. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*.

Le, L. D.; Chen, T.; Sadiq, S.; Yin, H.; Jin, M.; and Yu, J. 2026. Towards Efficient Reasoning in LLM-Based Recommender Systems via Model Merging. *arXiv preprint arXiv:2608.10447*.

Levit, M.; Ledgard, J.; Dong, H.; Suryanarayanan, V.; Kolman, E.; Tan, S.; Gan, Q.; and Chowdhary, V. 2026. Privacy-Preserving Data Drift Detection and Recovery for Large-Scale LLM Applications via Proxy Representations. *arXiv preprint arXiv:2608.08245*.

Li, Z. 2026. Beyond Routing: Decoupling Expert Dispatch and Aggregation in Sparse Mixture-of-Experts. *arXiv preprint arXiv:2608.08853*.

Liao, B.; Yu, C.; Yang, Y.; Wang, C.; Song, Z.; Liu, P.; and Wu, T. 2026. Path Integral Value Matching for Linear Quadratic Stochastic Optimal Control. *arXiv preprint arXiv:2608.10777*.

Lin, Y.; Ang, E.; Zhu, E.; Ding, B.; and Zhou, J. 2026. Context as an Environment: Programmatic Context Management for Long-Horizon Agents. *arXiv preprint arXiv:2608.21690*.

Liu, C.; Thiamwong, L.; Fu, Y.; and Xie, R. 2026a. Adaptive Multi-Agent Feature Selection for Personalized Fall Risk Prevention. *arXiv preprint arXiv:2608.18450*.

Liu, X.; Fang, Z.; He, H.; Hong, Y.; and Liu, D. 2026b. Triple Expert Learning from Noisy Labels for Semi-Supervised Vision Foundation Model Adaptation. *arXiv preprint arXiv:2608.09052*.

Liu, Y.; Lee, N.; and Anumanchipalli, G. K. 2026. StyleStream 2.0: Fast and Controllable Streaming Voice Style Conversion. *arXiv preprint arXiv:2610.28940*.

Liu, Z.; Huang, L.; Su, Z.; Pan, Z.; Han, Z.; Chen, X.; Song, Y.; and Zhang, W. 2026c. SkillLens: Visual Skill Cards for Retrieval-Augmented GUI Action Prediction and On-Policy Distillation. *arXiv preprint arXiv:2608.10775*.

Liu, Z.; Tian, Z.; Wang, X.; Wen, Z.; Xiong, Y.; Lai, Z.; and Li, D. 2026d. Efficient RLVR Scheduling via Graph-Structured Online Difficulty Estimation. *arXiv preprint arXiv:2608.17941*.

Lu, C.; Lu, C.; Lange, R. T.; Foerster, J.; Clune, J.; and Ha, D. 2024. The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery. *arXiv preprint arXiv:2408.06292*.

Luo, X.; Rechardt, A.; Sun, G.; Nejad, K. K.; Yáñez, F.; Yilmaz, B.; Lee, K.; Cohen, A. O.; Borghesani, V.; Pashkov, A.; et al. 2025. Large Language Models Surpass Human Experts in Predicting Neuroscience Results. *Nature Human Behaviour*, 9: 305–315.

Marcuzzi, F.; Christoph, M.; Gurevych, I.; and Rozovskays, A. 2026. Investigating the Impact of Learner Errors on Social Bias in Large Language Models. *arXiv preprint arXiv:2610.29896*.

Meng, Z.; Meng, Z.; Liu, S.; Xu, H.; Yuan, K.; and Ounis, I. 2026. SSE-Bio: A Structured Self-Evolving Agent with Agentic Retrieval Policy for Multi-Hop Biomedical Reasoning. *arXiv preprint arXiv:2608.22132*.

Miao, Q.; Guo, Y.; Zhang, L.; Cheng, Q.; Xiao, Y.; Ni, Z.; and Yang, S. 2026. Causal Spatiotemporal State Evolution for Monocular 4D Reconstruction. *arXiv preprint arXiv:2610.27594*.

Mule, S. P.; Garikaparthi, A.; and Patwardhan, M. 2026. Teaching Language Models to Forecast Research Success Through Comparative Idea Evaluation. *arXiv preprint arXiv:2605.21491*.

Nogueira, R.; Bonás, G. K.; Almeida, T. S.; Roque, A.; Pires, R.; Abonizio, H.; Laitz, T.; Larcher, C.; Malaquias Junior, R.; and Piau, M. 2026. Measuring Opinion Bias and Sycophancy via LLM-based Persuasion. *arXiv preprint arXiv:2604.21564*.

Novikov, A.; Vũ, N.; Eisenberger, M.; Dupont, E.; Huang, P.-S.; Wagner, A. Z.; Shirobokov, S.; Kozlovskii, B.; Ruiz, F. J. R.; Mehrabian, A.; et al. 2025. AlphaEvolve: A Coding Agent for Scientific and Algorithmic Discovery. *arXiv preprint arXiv:2506.13131*.

OpenAI. 2026a. GPT-5.6 System Card. https://deploymentsafety.openai.com/gpt-5-6. July 9, 2026.

OpenAI. 2026b. GPT-6 Astra System Card. https://deploymentsafety.openai.com/gpt-6-astra. September 3, 2026.

OpenAI. 2026c. GPT-6 Sol and GPT-6 Luna: October 2026 Update. https://deploymentsafety.openai.com/gpt-6-october. October 7, 2026.

Ozcam, B.; Onen, I.; Amasyali, M. F.; and Tatli, E. I. 2026. Generating Attacks for LLMs with GFlowNets. *arXiv preprint arXiv:2608.10171*.

Park, J.; Mendes, E.; Stanovsky, G.; and Ritter, A. 2025. Anticipatory Evaluation of Language Models: Predicting LLM Benchmark Scores from Descriptions. *arXiv preprint arXiv:2509.20645*.

Qu, G.; Li, J.; Xu, X.; Li, X.; Qin, B.; Huo, N.; Tang, Z.; Ma, C.; and Cheng, R. 2026. BIRD-RL: Scaling Agentic Reinforcement Learning over Stateful Data-Centric Environments. *arXiv preprint arXiv:2610.29426*.

Ravuru, C.; and Srivastava, S. 2026. UNMASK: Discovering and Causally Verifying Spurious Shortcuts in Text Classifiers. *arXiv preprint arXiv:2608.09209*.

Reflection AI. 2026. Introducing Beam: Reflection's 501B Open-Weight Model. https://reflection.ai/blog/introducing-beam. October 5, 2026.

Saha, D.; Mannan, S. M. A.; Rashid, M. R.; Naswan, R.; and Tahmid, A. 2026. Zero-Shot Traffic Accident Detection via a Coarse-to-Fine VLM-Tracking Pipeline. *arXiv preprint arXiv:2608.08867*.

Sajja, B.; Kleiman-Weiner, M.; Zimmermann, R.; and Zhi-Xuan, T. 2026. Evaluating Rational Contracting in Natural Language. *arXiv preprint arXiv:2608.10475*.

Sehwag, U. M.; Lau, E.; Oskouie, H. E.; Shabihi, S.; Liang, E.; Toledo, A.; Mangialardi, G.; Fonrouge, S.; Cardona, E.-Y. H.; Vergara, P.; et al. 2026. SciPredict: Can LLMs Predict the Outcomes of Scientific Experiments in Natural Sciences? *arXiv preprint arXiv:2604.10718*.

Sharma, M.; Tong, M.; Korbak, T.; Duvenaud, D.; Askell, A.; Bowman, S. R.; Cheng, N.; Durmus, E.; Hatfield-Dodds, Z.; Johnston, S. R.; Kravec, S.; Maxwell, T.; McCandlish, S.; Ndousse, K.; Rausch, O.; Schiefer, N.; Yan, D.; Zhang, M.; and Perez, E. 2024. Towards Understanding Sycophancy in Language Models. In *International Conference on Learning Representations (ICLR)*.

Shatri, E.; Mayer, J.; and Hajič, J. 2026. Data-Driven Glyph Positioning for Handwritten Music Synthesis. *arXiv preprint arXiv:2610.28899*.

Shen, Y.; and Unberath, M. 2026. A Multi-Agent Framework for Automated Machine Learning with Evidence-based Planning for Medical Image Analysis. *arXiv preprint arXiv:2610.28410*.

SpaceXAI. 2026. Introducing Grok 4.6. https://x.ai/news/grok-4-6. August 12, 2026.

Speck, Z.; and Shepard, A. 2026. Learned, Then Lost: A Measured Single-Example Counterfactual in Pre-training. *arXiv preprint arXiv:2608.19168*.

Sui, H.; Qin, S.; Liao, J.; Jia, X.; Chen, B.; and Liu, Y. 2026. ElasticBack: Stealthy Conditional Backdoor in LLM-Agent Skills via Coupled Trigger-Rule Optimization. *arXiv preprint arXiv:2608.09577*.

Talaei, S.; Daneshfar, F.; Abdullah, A. A.; and Oussalah, M. 2026. Reliability-Guided Vision-Language Decoding with Text-Aware Skip Refinement for Medical Image Segmentation. *arXiv preprint arXiv:2610.28118*.

Tan, S.; Zhou, B.; Agrawal, L. A.; Zhang, J.; Chen, Y.; Mang, Q.; Lee, D.; Jacob, D.; Sen, K.; Dimakis, A.; and Zaharia, M. 2026. Scaling Discovery Requires Forgetting. *arXiv preprint arXiv:2610.20563*.

Thinking Machines Lab. 2026. Inkling Model Card. https://huggingface.co/thinkingmachines/Inkling. July 15, 2026.

Tian, Q.; Yin, H.; Su, H.; Chao, J.; Gu, X.; Kong, Y.; Xia, Y.; and Liu, Z. 2026. ForeSci: Evaluating LLM Agents for Forward-Looking AI Research Judgment. *arXiv preprint arXiv:2606.00644*.

Topf, S.; Sanchez-Fernandez, A.; and Klambauer, G. 2026. BatchBench: Benchmarking AI Robustness to Batch Effects and Domain Shift in Biomedical Images - A JUMP Cell Painting Leaderboard. *arXiv preprint arXiv:2610.30104*.

Velichko, A.; and Mahmoud, I. 2026. Asymmetric Bidirectional Predictive Dynamics in Google Search and COVID-19 Time Series: A Multi-Horizon Forecast-Error Analysis. *arXiv preprint arXiv:2610.28563*.

Wang, C.; Wang, L.; Pan, J.; Wang, W.; Ren, W.; Liu, J.; and Cao, X. 2026a. Ultra-High-Definition Restoration Transformers with Correlation Matching Transformation. *arXiv preprint arXiv:2608.20263*.

Wang, H.; Zhu, Y.; Wang, T.; Li, Y.; Dong, J.; Zhang, X.; and Ren, L. 2026b. Physics-informed Diffusion Generative Model for Time-Series Data Synthesis in Dynamic Systems. *arXiv preprint arXiv:2608.10941*.

Wang, J.; Wang, Z.; Guo, Y.; Yang, Y.; and Liang, Y. 2026c. TRACE-Memory: Public-Conditioned Retrieval and Utility-Aware Evidence Admission for Personalized Generation. *arXiv preprint arXiv:2608.08446*.

Wang, Z.; Nie, Y.; Ma, J.; and Li, Y. 2026d. Beyond the Vote: When Do LLMs Add Value to Retrieval-Based Tabular Classification? *arXiv preprint arXiv:2610.29520*.

Wen, J.; Si, C.; Chen, Y.-H.; He, H.; and Feng, S. 2025. Predicting Empirical AI Research Outcomes with Language Models. *arXiv preprint arXiv:2506.00794*.

Weng, Y.; Zhu, M.; Xie, Q.; Sun, Q.; Lin, Z.; Liu, S.; and Zhang, Y. 2025. DeepScientist: Advancing Frontier-Pushing Scientific Findings Progressively. *arXiv preprint arXiv:2509.26603*.

Wijk, H.; Lin, T.; Becker, J.; Jawhar, S.; Parikh, N.; Broadley, T.; Chan, L.; Chen, M.; Clymer, J.; Dhyani, J.; et al. 2024. RE-Bench: Evaluating Frontier AI R&D Capabilities of Language Model Agents Against Human Experts. *arXiv preprint arXiv:2411.15114*.

Wu, S.; Lu, P.; Chen, Y.; Bragg, J.; Yamada, Y.; Clark, P.; Clifton, D.; Torr, P.; Zou, J.; and Yu, J. 2026. Scientific Reasoning Does Not Reliably Translate into Scientific Forecasting in Frontier AI. *arXiv preprint arXiv:2605.22681*.

Xu, W.; Fan, J.; Taylor, E.; Wang, Z.; Joseph, R.; and Kong, F. 2026. SLoSa: Small, Low-Power, and Safe Foundation Models for Robotic Task Planning. *arXiv preprint arXiv:2610.28955*.

Yamamoto, K.; and Okusa, K. 2026. MiCLA: Mixture-to-Constituent Latent Alignment for Polyphonic Audio Representation Learning. *arXiv preprint arXiv:2610.27491*.

Yen, Y.-F. 2026. Decomposition-Induced Context-Memory Conflict: When Fact-Checking Pipelines Contradict Their Own Source Text. *arXiv preprint arXiv:2608.10627*.

Yu, D.; Zhao, Q.; Yue, Y.; Li, Y.; Zhang, Z.; and Zheng, G. 2026. UNIVERS: Learning UNIversal Voxel-wise Embedding via Randomized Synthesis for generalizable one-shot landmark localization. *arXiv preprint arXiv:2610.29098*.

Z.ai. 2026a. GLM-5.3-Flash Model Card. https://huggingface.co/zai-org/GLM-5.3-Flash.

Z.ai. 2026b. GLM-5.3 Model Card. https://huggingface.co/zai-org/GLM-5.3.

Zamora, C.; Zuniga, H.; Orozco-Rosas, U.; and Picos, K. 2026. Retrieval-Augmented Vision Foundation Models for Robust Leukemia Cell Classification across Multiple Microscopy Datasets. *arXiv preprint arXiv:2608.10657*.

Zechner, M. 2026. pi: a minimal coding agent. https://github.com/badlogic/pi-mono. Npm package `@earendil-works/pi-coding-agent`, version 1.1.0.

Zhang, A. L.; Kraska, T.; and Khattab, O. 2025. Recursive Language Models. *arXiv preprint arXiv:2512.24601*.

Zhao, Y.; Pandey, S.; and Misra, P. K. 2026. What the Author Wrote, What the Agent Reads: Silent Cue Loss in Enterprise Document Ingestion. *arXiv preprint arXiv:2610.30026*.

Zheng, J.; Zhang, J.; Luo, Y.; Mao, Y.; Gao, Y.; Du, L.; Chen, H.; and Zhang, N. 2026a. Can We Predict Before Executing Machine Learning Agents? *arXiv preprint arXiv:2601.05930*.

Zheng, Y.; Huang, W.; Ma, M.; Jin, K.; and Jiang, Y. 2026b. Closed-Loop Dual-Reference Guidance for Accent Control in Cross-Lingual Voice Cloning. *arXiv preprint arXiv:2610.29351*.

Zou, Q.; Yuan, B.; Song, S.; Liu, W.; and Wang, X. 2026. The Predictive–Optimization Gap in GuanDan Self-Play. *arXiv preprint arXiv:2610.29411*.

Zúñiga, N. V. 2026. What Iterated Self-Feeding Probes of Language Models Measure, and a test that separates the construction from the model. *arXiv preprint arXiv:2608.10986*.

Appendix A: Retrieval Pipeline

We draw arXiv e-print LaTeX sources from a fixed submission window and parse them into sections and tables. Five LLM stages then run on each paper, called through OpenRouter and, for the October batch's enumeration and composition stages, through Hugging Face Inference Providers. A cheap gate reads only the title, abstract, section outline, and table captions and answers three yes/no questions: is this AI research, does it introduce something of its own, and are results reported. An enumerator lists every controlled comparison, meaning two variants that differ in one technique, as a short description with its benchmarks and metrics but no numbers. The paper is kept outside the model's context window, as in recursive language models (Zhang, Kraska, and Khattab 2025): a root model can grep it, read a page, or send named sections to a subagent, receives one-line digests of what the subagents found, and may not finish until every section and table is covered or skipped with a stated reason. A retrieval agent (OpenRouter slug `z-ai/glm-5.2`, and `z-ai/glm-5.3` for the October batch) then finds each comparison again and submits start and end anchors instead of text; Python copies the exact source bytes between them for the explanatory passages and the result rows, and both the baseline and the method cell must appear literally in a copied row, so nothing in the payload is model-written and "not found" is an accepted answer. A composer that never sees the method values rewrites the copied passages into a first-person researcher message ending with the baseline numbers; a mechanical check rejects any number, LaTeX command, or leaked method value that is not in the sources, and a judge model, also blind to the method values, then checks the message for hints of the outcome and for fidelity to the passages. Python appends the question, which asks for the value of each (benchmark, metric) pair after switching from the baseline arm to the method arm, and the graded target is that measurement grid. These checks guarantee verbatim text and numbers, not that the two cells form the right comparison, so every sample was manually audited against its paper afterwards. The system prompt of every stage is in the pipeline repository, verbatim as run: the eligibility gate in `arl/extract.py`, the enumerator's root and subagent prompts in `arl/candidates.py`, the retrieval agent in `arl/retrieve.py`, the composer in `arl/compose.py` and the judge in `arl/verify.py`. Model slugs, verbatim from the run records and the pipeline configuration: stage 4 `~deepseek/deepseek-v4-flash-latest`; stage 5 root and enumerators `anthropic/claude-sonnet-5`; stage 6 as above; stage 7 and stage 8 `anthropic/claude-sonnet-5`.

Appendix B: A Complete Sample

Sample `2608.09209-s051` (Ravuru and Srivastava 2026) from the evaluation set, reproduced verbatim from the corpus. The prompt is sent verbatim as the user turn, after a fixed system turn, the same for every sample and every model, that asks for one point estimate per slot in a closing JSON block. The target is the graded measurement grid, which the model never sees. This sample passed every automated check and was not hand-edited.

System turn

````
You are forecasting the outcome of a machine
-learning experiment that has already been
run, from the researcher's description of it
 alone. The results are not in front of you
and you cannot look them up; estimate them.

Think it through in as much depth as you
want first. Then close your reply with a
single fenced JSON block, with nothing after
 it:

```json
{"predictions": [{"slot": "<label, copied
exactly>", "value": <number>}]}
```

Rules for that block:

1. One entry per metric the researcher asks
you to predict, in the order they are asked,
 and no extra entries.
````

```
2. "slot" is the metric's label copied
exactly as it appears in the question,
including any parenthesised unit.
3. "value" is a bare JSON number on that
metric's own scale -- the same scale the
researcher quoted their baseline on. No
percent signs, no units, no thousands
separators, no ranges, no strings, no nulls.
4. Commit to a single best point estimate
for every metric. Predicting no change is a
legitimate answer; refusing to answer is not
.

The prose before the block is yours to use
however you like. Only the block is read
mechanically.
```

Prompt (user turn)

```
I've been spending cycles on isolating how
much of the gain in our BERT DFR debiasing
pipeline actually comes from the discovery
stage versus the downstream recipe. The idea
 is to hold the debiasing recipe fixed and
only vary how candidate features/groups are
discovered, comparing a few discovery
baselines: PMI (top-10 single-token
statistics), PMI+SFV (PMI candidates passed
through our statistical validator), and LLM-
only (prompt-only LLM candidates with no
validation funnel). The point is to check
whether cheaper feature mining could produce
 the same groups as the full pipeline, which
 would mean the discovery stage isn't
earning the gains on its own. My setup is
BERT-base-uncased with DFR on CivilComments-
WILDS, run across 3 seeds, comparing the
full UNMASK discovery pipeline against the
LLM-only variant (no statistical/causal
validation) under this fixed downstream
recipe. My baseline gets, for CivilComments-
WILDS with the LLM-only discovery variant,
WGA = 65.1% and avg-acc = 89.71%.

My question about this setup: after
switching from LLM-only (my baseline above)
to UNMASK, predict the resulting value for
each: CivilComments-WILDS WGA (in %);
CivilComments-WILDS avg-acc (in %).
```

Target

```
{
  "measurements": [
    {
      "benchmark": "CivilComments-WILDS",
      "metric": "WGA",
      "baseline_value": 65.1,
      "method_value": 71.84,
      "unit": "%"
    },
    {
      "benchmark": "CivilComments-WILDS",
      "metric": "avg-acc",
      "baseline_value": 89.71,
      "method_value": 90.99,
      "unit": "%"
    }
  ]
}
```

Appendix C: Research-Area Taxonomy

Each preprint carries one research-area tag, assigned by hand from its title and abstract and applied to every sample built from it. Table 12 gives the six areas together with the rule used to assign them; where a preprint straddles two areas the rule decides, so an attack on image watermarks (Cao et al. 2026b) is tagged safety rather than vision, model merging for an LLM recommender (Le et al. 2026) is tagged applied ML rather than LLM training, and a benchmark whose subject is a multi-agent system (Chen et al. 2026b) is tagged agents rather than evaluation. The six areas fall into two groups. For agents, safety, and evaluation studies the forecast concerns the behaviour of a language model or an LLM-based system; for vision, applied, and LLM-training work it concerns a trained model's score on a conventional benchmark. Table 13 lists and cites every corpus preprint with its tag and the number of corpus samples it contributes.

Appendix D: Additional Results

Table 15 repeats Table 7 with the median per-sample error in place of the mean. Table 16 extends Table 9 to all three failure modes.

Table 12: The six research areas and their definitions.

Area	Definition and assignment rule
Vision & multimodal	Vision or vision-language models on vision tasks: classification, segmentation, restoration, counting, detection, token pruning in ViTs, medical imaging, report generation. Rule: the input modality is images or video and the metric is a vision benchmark score.
Safety, attacks, interpretability & auditing	Studies that probe, attack, audit or measure a model's behaviour or internals rather than improve a task score: jailbreaks, backdoors, watermark removal, attack generation, probes and their transfer, concept erasure, shortcut discovery, mechanistic measurement. Rule: the proposed method is adversarial, or the measured quantity is what the model internally does.
LLM agents, memory & context	An LLM agent is the system under study: skill libraries, memory and retrieval policies, context management, GUI or knowledge-work agents, multi-agent orchestration, LLM agents in games. Rule: the thing being measured is an agent loop, even when the paper is framed as a benchmark.
Applied & scientific ML	A learned model applied to a non-LLM scientific or engineering problem: acoustic localisation, stochastic control, time-series conformal prediction, reliability test planning, computational neuroscience, clinical risk prediction, recommender systems. Rule: the metric is a domain quantity (metres, control error, interval width), not an LLM behaviour.
LLM training & architecture	Pre-training, post-training and architecture of language models themselves: nested model suites, self-distillation, RL scheduling, pre-training counterfactuals, MoE routing. Rule: the intervention changes how a language model is trained or built, and the metric is a standard LM score (perplexity, accuracy, cross-entropy).
LLM evaluation studies	How LLMs are evaluated or how their use is measured, with no agent loop involved: benchmark construction, judge calibration and drift, usage-rate measurement, cost and latency of reasoning modes. Rule: the paper is about the evaluation apparatus or a population measurement, not a model or method.

Table 13: Area assignment of all 71 corpus preprints, with the number of corpus samples each contributes.

Area	arXiv	Title	Samples
Vision & multimodal	2608.08867	Zero-Shot Traffic Accident Detection via a Coarse-to-Fine VLM-Tracking Pipeline (Saha et al. 2026)	2
	2608.09052	Triple Expert Learning from Noisy Labels for Semi-Supervised Vision Foundation Model Adaptation (Liu et al. 2026b)	8
	2608.09656	EgoHieraLoc: A Cortically Inspired Hierarchical Segmentation-Guided Framework for Egocentric Visual Query Localization (Cao et al. 2026c)	22
	2608.10544	Flow Straight to Reality: Perceptually Consistent Flow Matching for Efficient Image Restoration (Jo et al. 2026)	10
	2608.10657	Retrieval-Augmented Vision Foundation Models for Robust Leukemia Cell Classification across Multiple Microscopy Datasets (Zamora et al. 2026)	3
	2608.10989	Putting Registers to Work: Task Registers for Token Pruning in Vision Transformers (Cao et al. 2026a)	24
	2608.20263	Ultra-High-Definition Restoration Transformers with Correlation Matching Transformation (Wang et al. 2026a)	22
	2608.20621	RECOUNT: Reference-guided Counting with Synthetic Visual Exemplars (D'Alessandro, Mahdavi-Amiri, and Hamarneh 2026)	9
	2608.23143	An end-to-end-trained vision-language model for native-language prostate pathology report generation (Grashei et al. 2026)	1
	2610.27594	Causal Spatiotemporal State Evolution for Monocular 4D Reconstruction (Miao et al. 2026)	3
	2610.28118	Reliability-Guided Vision-Language Decoding with Text-Aware Skip Refinement for Medical Image Segmentation (Talaei et al. 2026)	2
	2610.28423	WorldSR: Harnessing World Knowledge Search for Grounded Image Super-Resolution (Deng et al. 2026)	10
	2610.28630	SolarRenderer: Few-Step Generative Reference Rendering (Fang et al. 2026)	10
	2610.28899	Data-Driven Glyph Positioning for Handwritten Music Synthesis (Shatri, Mayer, and Hajič 2026)	5
	2610.29098	UNIVERS: Learning UNIversal Voxel-wise Embedding via Randomized Synthesis for generalizable one-shot landmark localization (Yu et al. 2026)	2
	2610.30104	BatchBench: Benchmarking AI Robustness to Batch Effects and Domain Shift in Biomedical Images - A JUMP Cell Painting Leaderboard (Topf, Sanchez-Fernandez, and Klambauer 2026)	2
Safety, attacks, interpretability & auditing	2608.09209	UNMASK: Discovering and Causally Verifying Spurious Shortcuts in Text Classifiers (Ravuru and Srivastava 2026)	25
	2608.09551	Pragmatic Attack Surface: Vulnerabilities of Implicit Context in Large Language Models (Chen et al. 2026a)	9
	2608.09577	ElasticBack: Stealthy Conditional Backdoor in LLM-Agent Skills via Coupled Trigger-Rule Optimization (Sui et al. 2026)	2
	2608.10166	MarkNull: Model-Agnostic Watermark Removal in AI-Generated Images via On-Manifold Latent Manipulation (Cao et al. 2026b)	4
	2608.10171	Generating Attacks for LLMs with GFlowNets (Ozcam et al. 2026)	1
	2608.10566	Iterative Erasure Count Is Not an Affine-Invariant Concept Dimension (Jin et al. 2026b)	8
	2608.10627	Decomposition-Induced Context-Memory Conflict: When Fact-Checking Pipelines Contradict Their Own Source Text (Yen 2026)	3
	2608.10986	What Iterated Self-Feeding Probes of Language Models Measure, and a test that separates the construction from the model (Zúñiga 2026)	4
	2608.19338	Mechanistic Tomography: Designed Measurement for Control-Oriented Interpretability (Erramilli 2026)	5
	2610.27304	Global Security Blind Spots: Translation as a Safety Bypass in Low-Resource Languages (Alva et al. 2026)	4
	2610.29342	Uncensored Is Not Unbiased: Abliteration Removes Refusals but Not Script-Dependent Bias About Taiwan (Ko 2026)	11
	2610.29896	Investigating the Impact of Learner Errors on Social Bias in Large Language Models (Marcuzzi et al. 2026)	2
LLM agents, memory & context	2608.08037	SkillSmith: Enhancing Locally Deployed Agents via Automatic Skill Construction and Evolution (Jiang, Xie, and Tang 2026)	8
	2608.08446	TRACE-Memory: Public-Conditioned Retrieval and Utility-Aware Evidence Admission for Personalized Generation (Wang et al. 2026c)	6
	2608.08605	ForestBench: A Unified Graph Framework for Evaluating Multi-Agent Collaboration (Chen et al. 2026b)	2
	2608.10475	Evaluating Rational Contracting in Natural Language (Sajja et al. 2026)	1
	2608.10775	SkillLens: Visual Skill Cards for Retrieval-Augmented GUI Action Prediction and On-Policy Distillation (Liu et al. 2026c)	3
	2608.13940	AI Research Preference Models (Foster et al. 2026)	6
	2608.18050	StagedWorkspace: A Versioned Workspace for Knowledge-Work Agents (Hua et al. 2026)	4
	2608.21690	Context as an Environment: Programmatic Context Management for Long-Horizon Agents (Lin et al. 2026)	1

Table 14: Area assignment of all 71 corpus preprints (continued).

Area	arXiv	Title	Samples
LLM agents, memory & context	2608.22132	SSE-Bio: A Structured Self-Evolving Agent with Agentic Retrieval Policy for Multi-Hop Biomedical Reasoning (Meng et al. 2026)	2
	2610.07447	Fork-and-Flush: Escaping Idea Basins in Autoresearch Agents (Cai et al. 2026)	6
	2610.20563	Scaling Discovery Requires Forgetting (Tan et al. 2026)	4
	2610.27480	Runtime Control for AI Agents: Closing the Evaluation–Execution Gap in Operational Workflows (Bhadane et al. 2026)	1
	2610.28410	A Multi-Agent Framework for Automated Machine Learning with Evidence-based Planning for Medical Image Analysis (Shen and Unberath 2026)	4
	2610.29426	BIRD-RL: Scaling Agentic Reinforcement Learning over Stateful Data-Centric Environments (Qu et al. 2026)	3
	2610.30026	What the Author Wrote, What the Agent Reads: Silent Cue Loss in Enterprise Document Ingestion (Zhao, Pandey, and Misra 2026)	5
Applied & scientific ML	2608.08911	Physics-Informed Learning for Robust Acoustic Localization with Calibrated Uncertainty (Kampe et al. 2026)	3
	2608.09622	Adaptive Sequential Test Planning for Multi-Mechanism Reliability Qualification via Bayesian Monte Carlo Tree Search (Elhagrasy, Hill, and Ivanov 2026)	10
	2608.10447	Towards Efficient Reasoning in LLM-Based Recommender Systems via Model Merging (Le et al. 2026)	14
	2608.10553	Retrieval-Corrected Conformal Prediction for Time Series (Jin et al. 2026a)	12
	2608.10777	Path Integral Value Matching for Linear Quadratic Stochastic Optimal Control (Liao et al. 2026)	6
	2608.10941	Physics-informed Diffusion Generative Model for Time-Series Data Synthesis in Dynamic Systems (Wang et al. 2026b)	1
	2608.18450	Adaptive Multi-Agent Feature Selection for Personalized Fall Risk Prevention (Liu et al. 2026a)	16
	2608.18569	The Role of Grid Cells in Reducing Spatial Aliasing in Hippocampal Place Representations (Johnson, Ghizawi, and Minai 2026)	1
	2610.27491	MiCLA: Mixture-to-Constituent Latent Alignment for Polyphonic Audio Representation Learning (Yamamoto and Okusa 2026)	6
	2610.28178	Coincidence-Maximizing Customer Clustering Using Smart Meter Data (Gálvez, Venturini, and Mondrzak 2026)	4
	2610.28563	Asymmetric Bidirectional Predictive Dynamics in Google Search and COVID-19 Time Series: A Multi-Horizon Forecast-Error Analysis (Velichko and Mahmoud 2026)	3
	2610.28940	StyleStream 2.0: Fast and Controllable Streaming Voice Style Conversion (Liu, Lee, and Anumanchipalli 2026)	3
	2610.28955	SLoSa: Small, Low-Power, and Safe Foundation Models for Robotic Task Planning (Xu et al. 2026)	5
	2610.29351	Closed-Loop Dual-Reference Guidance for Accent Control in Cross-Lingual Voice Cloning (Zheng et al. 2026b)	5
	2610.29411	The Predictive–Optimization Gap in GuanDan Self-Play (Zou et al. 2026)	1
	2610.29589	Fragment-Level Graph Encoding for Interpretable Molecular Property Prediction (Almeida and Pöhner 2026)	4
	2610.29605	Improved Training of Strong Stochastic Flow Maps (Blasingame et al. 2026)	14
LLM training & architecture	2608.08853	Beyond Routing: Decoupling Expert Dispatch and Aggregation in Sparse Mixture-of-Experts (Li 2026)	7
	2608.09228	Privileged Solutions or Context-Induced Teacher Behavior? Dissecting On-Policy Self-Distillation (Ichihara et al. 2026)	29
	2608.09703	Matryoshka Language Model Suites (Godey and Artzi 2026)	6
	2608.17941	Efficient RLVR Scheduling via Graph-Structured Online Difficulty Estimation (Liu et al. 2026d)	11
	2608.19168	Learned, Then Lost: A Measured Single-Example Counterfactual in Pre-training (Speck and Shepard 2026)	3
	2610.29419	When Thoughts Become Memory: Unifying In-Context Learning and Fine-Tuning (Chlon and Huang 2026)	15
LLM evaluation studies	2608.08245	Privacy-Preserving Data Drift Detection and Recovery for Large-Scale LLM Applications via Proxy Representations (Levit et al. 2026)	3
	2608.10715	Most biomedical publications show signs of LLM-assisted writing (Holzwarth, González-Márquez, and Kobak 2026)	1
	2608.20204	ContractScrub: A benchmark for final review of legal contracts (Bang et al. 2026)	2
	2610.27868	Hallucinations of Large Language Models in Cross-Language Code Translation: Phenomena and Mitigation (Du, Zhang, and Li 2026)	2
	2610.29520	Beyond the Vote: When Do LLMs Add Value to Retrieval-Based Tabular Classification? (Wang et al. 2026d)	1

Table 15: Median error by research area.

Model Samples	Vision 16	Safety 12	Agents 15	Applied 17	LLM-train 6	LLM-eval 5	All 71
Grok 4.6	0.06	0.30	**0.15**	0.29	0.14	**0.08**	**0.12**
GPT-6 Astra	0.05	0.17	0.16	0.17	**0.10**	0.12	0.13
Claude Opus 5.5	0.06	0.15	0.23	0.17	0.13	0.09	0.13
Claude Fable 5.1	0.04	0.22	0.16	0.14	0.13	0.11	0.13
Kimi K3	0.07	0.34	**0.15**	0.20	0.14	0.14	0.14
GPT-6 Sol	0.05	0.31	0.21	0.20	0.13	0.12	0.14
DeepSeek V4 Flash	0.09	0.38	0.19	0.12	0.13	0.14	0.14
GLM-5.3	0.05	0.34	**0.15**	0.16	0.14	0.14	0.14
Gemini 4 Argon	0.05	0.23	0.22	0.16	0.19	0.11	0.15
GPT-5.6 Sol	**0.03**	0.28	0.17	0.18	0.12	0.13	0.15
GPT-6 Luna	0.04	0.39	**0.15**	0.15	0.14	0.14	0.15
Inkling	0.06	0.28	0.25	**0.07**	0.15	0.15	0.15
DeepSeek V4 Pro	0.06	0.24	0.30	0.17	0.15	0.13	0.15
Claude Fable 5	0.10	**0.14**	0.28	0.28	0.12	0.12	0.16
Beam 1.0	0.06	0.19	0.23	0.27	0.13	0.16	0.17

Table 16: All failure modes by model.

Model	Traces	Mode 1			Mode 2			Mode 3			None
		n	with	without	n	with	without	n	with	without	
GPT-6 Luna	71	5	0.12	0.29	9	0.49	0.25	**0**	–	0.28	**58**
GPT-6 Astra	71	7	0.33	0.25	7	0.46	0.23	3	0.22	0.26	56
Claude Opus 5.5	69	**2**	0.19	0.26	**5**	0.58	0.23	9	0.27	0.25	54
GPT-6 Sol	71	5	0.21	0.28	7	0.44	0.25	6	0.16	0.28	53
GPT-5.6 Sol	71	11	0.27	0.27	11	0.43	0.24	6	0.55	0.24	46
Beam 1.0	71	11	0.41	0.27	14	0.54	0.23	14	0.51	0.24	43
Gemini 4 Argon	68	7	0.38	0.25	11	0.42	0.24	17	0.28	0.26	41
DeepSeek V4 Flash	71	10	0.35	0.26	13	0.50	0.22	18	0.34	0.25	41
GLM-5.3	71	6	0.18	0.28	14	0.55	0.21	19	0.36	0.24	41
Claude Fable 5.1	67	3	0.51	0.25	8	0.57	0.22	18	0.20	0.28	39
Claude Fable 5	66	3	0.30	0.26	7	0.48	0.24	25	0.30	0.24	35
Inkling	70	18	0.40	0.25	16	0.57	0.20	23	0.40	0.23	35
Kimi K3	70	12	0.29	0.25	12	0.56	0.20	37	0.30	0.21	29
DeepSeek V4 Pro	70	11	0.49	0.25	13	0.44	0.25	32	0.31	0.26	28
Grok 4.6	71	44	0.32	0.21	14	0.56	0.21	10	0.26	0.28	23
All	1048	155	0.33	0.26	161	0.51	0.23	237	0.32	0.26	622
Share		15%			15%			23%			59%

Appendix E: Autoresearch-Harness Brief

The brief below is copied verbatim from the arena at the time of writing. It is the task both arms receive; nothing in it names a benchmark item. The custom arm's extension prompts are in the released harness source (the Autoresearch-Harness section); the vanilla arm runs pi with its stock prompts only.

E.1 The brief

The file `BRIEF.md`, given to the researcher as its first prompt. `$INNER_API_BASE`, `$INNER_MODEL` and `$INNER_API_KEY` are environment variables set in the researcher container.

```
# Brief: build a long-horizon agent

You are a researcher. Your job is to build
an agent that can take a question
about a body of files, work through those
files with a shell
and Python over many steps, and commit to
one answer. The agent lives in
`agent/`. The agent is to be used for long
horizon reasoning tasks.

## What the agent is for

Long-horizon work over a workspace of files:
 investigations that take tens of
tool calls, where the agent has to find out
what is in the workspace, read the
documentation that explains it, follow a
chain of evidence across several
files, compute things along the way, check
its own intermediate results, and
arrive at a conclusion it can defend. The
workspaces are collections of data
files and their documentation; the questions
 are quantitative and the
documentation decides what the terms mean.
The agent is given the workspace
and the question when it is run, and nothing
 before that. Some questions have
no valid answer in the workspace, and
recognising that is part of the job.

Expect every task to be well beyond what a
single model turn can do. The tasks
are the hard kind: each needs many steps,
and none is a simple lookup.

## Conditions the agent runs under

- One fresh process per task, with no memory
 of previous tasks.
- Shell and Python 3.12, with pandas and
numpy available and nothing else
  installable. No network.
- A language model it can call at `
$INNER_API_BASE` with the name
  `$INNER_MODEL`, through an OpenAI-
compatible chat API with tool calling. This
  is the only model it has.

## Your budget

You have 4.00 US dollars of model calls for
the whole job: the calls your agent
makes to `$INNER_API_BASE` whenever you run
it, whether on a practice task you
built, on a quick check, or on anything else
. A single run of the agent on a
task of this genre costs on the order of a
cent or two. When the budget is
spent, the endpoint returns HTTP 402 and the
 job is over; whatever is in
`agent/` at that moment is your deliverable.
You can read the remaining budget at any
time:

    curl -s -H "authorization: Bearer
$INNER_API_KEY" http://gateway:8080/inner/
usage

We evaluate the deliverable afterwards on
our own budget.

## How you work

State lives in this repository: `results.tsv
` for your log, `practice/` for
anything you build to test with, `agent/`
for the deliverable. Your context may
be reset while you work, so keep the log
current enough to resume from. Write
`DONE` in the repository root when you judge
 that no further experiment is
worth its cost.

## What you start from

`agent/` contains a minimal working agent.
It is a starting point, not a
recommendation.

## Deliverable

The directory `agent/`, honouring the
command-line contract in
`agent/README.md` exactly: the flags, the
answer-file format, exit code 0. Run
the contract check described there before
you finish. An agent that fails it
scores zero on every task.
```