LEARNING WHAT TO REVISE AFTER A PREDICTION ERROR

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate sequential diagnosis, considering whether models can use new evidence to distinguish a faulty observation from a changed state or governing rule, and whether lightweight fine-tuning on synthetic examples can improve their ability to identify what revisions benefit real scenarios. We build matched triplets in which the three causes are indistinguishable until later evidence arrives, with an exact Bayesian reference at every step, and find that language models compute the diagnosis but do not report it. LOCUSPROBE, a linear probe on Llama-3.1-8B's hidden states, identifies the cause on real weather series with 0.97 accuracy against 0.44 for the same model's answers, and beats every tested open direct-answer system on matched items, including the nine-times-larger Qwen2.5-72B by 0.125 $[0.080, 0.173]$. Posterior-matched tuning (PMT) on synthetic arithmetic and logic alone transfers to real weather and industrial series with no real labels; applied to the unembedding only, with every other weight frozen, it raises rule-change recall from 0.03 to 0.72. When the diagnosis selects the revision, probe-selected revisions recover 98.5–99.9% of the gap to an oracle over 60 steps and attain a worst-case loss of 6.63 against 14.98 for an observer given the exact generative family of all three causes. Recall on rule changes, not overall accuracy, predicts that recovery ($\rho = -0.946$ versus -0.808), because an uncorrected rule compounds. These results identify a learnable component of revision, i.e., using subsequent evidence to determine what went wrong, what should change, and whether that revision allows the system to recover.

Code: https://anonymous.4open.science/r/locus-revise

1 INTRODUCTION

In time series prediction, an unexpected reading can draw from several sources and call for several different updates. Learning from a prediction error requires deciding whether to revise the observation, the current state, or the rule used to predict what happens next. At first, the model may have no way to tell these causes apart, yet each requires a different correction: fixing a bad reading, updating the current state, or changing the rule used for future predictions. For example, in predicting robot motion, a current sensor reading may result from a robot's position having been perturbed, or the sensor reporting a wrong value, or a movement command having an unexpected effect. Correcting the reading, revising the current location, or changing the motion rule will lead to different future predictions. Accommodating and recovering from a discrepancy therefore depends on which explanation the later evidence supports. The consequences of these mistakes can also differ over time. A faulty observation may affect only one reported value, whereas a changed rule can continue to alter every prediction that follows. Diagnosis accuracy can therefore hide an important distinction between errors that disappear and errors that compound. Throughout, we call the three causes OBS (a faulty observation), STATE (a disturbed state), and LAW (a changed rule).

Researchers have long studied how to explain system failures and update beliefs when the world changes (Willsky, 1976; Boutilier, 1998). Recently, Belief-R (Wilie et al., 2024) has tested whether language models change their conclusions when given new information, while BeliefTrack (Xu et al., 2026) has tested whether they keep track of which rules or faults remain possible as evidence arrives or is corrected. Here we test whether language models can tell what went wrong and learn to make the right correction. We further find that a linear readout trained only on synthetic

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.

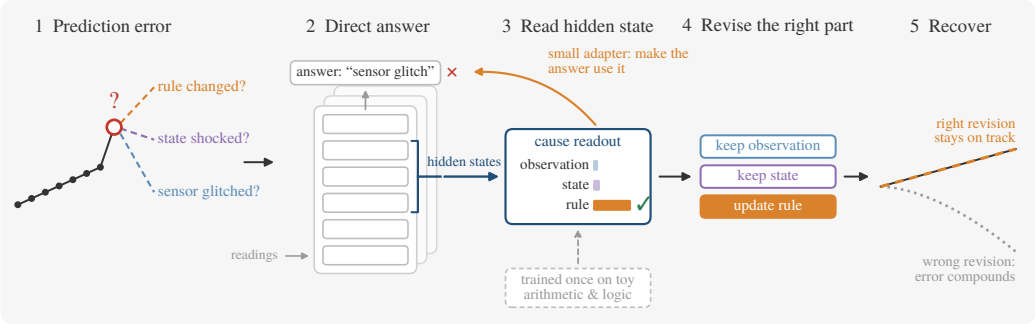


Figure 1: The same surprising reading can come from a faulty observation, a changed state, or a changed rule. A language model's answer names the wrong cause, while a probe on its hidden state finds the right one; training on synthetic arithmetic and logic makes the answer use it, and the right revision keeps the forecast on track.

arithmetic and logic representations of Llama-3.1-8B transfers to real weather and industrial data without target-domain labels. Finally, when each diagnosis is used to revise a forecasting system, long-horizon recovery depends strongly on which cause the model gets right, with LAW recall more closely associated with 60-step error than overall cause accuracy. To this end, we have configured a dataset in which each example has three versions: one with a faulty reading, one with a changed state, and one with a changed rule. All three initially show the model exactly the same information, so it cannot know the cause until further observations reveal the difference. We find that a trained classifier can identify the cause from a model's hidden states much more accurately than the model reports it directly. We also show that fine-tuning on synthetic examples produces substantial gains in cause identification on real time series data, and demonstrate that choosing the correct cause improves subsequent actions. These experiments leverage what the evidence reveals, what can be recovered from the model's hidden states, and what the model can learn to report and use to improve time series prediction. The research question that this work addresses is:

In time series prediction, can LLMs identify the cause of faulty data, and if so, can this information help improve the quality of future predictions?

To this end, we introduce a benchmark dataset containing 600 groups of three examples across arithmetic, logic, and navigation sequences, with each group showing how the same initial error can arise from three different causes: an incorrect observation, a change in the underlying state, or a change in the rule governing the system. The model initially sees the same history and the same unexpected observation in all three examples, so it cannot yet distinguish the causes and should therefore assign each a probability of $1/3$. We then provide further observations that differ between the examples and reveal their respective causes, allowing us to test whether models remain uncertain while information is insufficient and subsequently identify the correct cause once enough evidence arrives.

We show that training on simple synthetic examples can improve diagnoses on real time series, raising cause-identification accuracy from 0.307 to 0.597 on NOAA-based weather data and from 0.428 to 0.580 on SKAB-based industrial data. To test models on failures that actual systems experienced, we also use four existing datasets containing recordings from a water-circulation testbed (SKAB), Spark computing clusters (Exathlon), an industrial-control testbed (HAI 21.03), and aircraft flights with failures (ALFA) (Katser & Kozitsin, 2020; Jacob et al., 2021; Shin et al., 2021; Keipour et al., 2019). We compare these recorded failures with artificial errors added to data from the same sources and find that models generally perform better on the artificial errors, with higher accuracy in 32 of 44 model–dataset comparisons after accounting for differences in cause frequencies, and statistically significant differences in 18. Our results therefore show that synthetic training can teach useful diagnostic skills, notwithstanding evidence of a sim-to-real gap when applied to real data.

The studies yield four connected findings. First, there is a gap between the causal information a trained probe can recover from a model's hidden states and the accuracy of the model's direct

answers. On NOAA-based items, Llama-3.1-8B reaches 0.973 probe accuracy versus 0.438 direct-answer accuracy. A stricter cross-domain test shows that this information can also transfer: a linear probe trained only on arithmetic and logic hidden states transfers to real weather and industrial series without target-domain labels, where classifiers on the input itself stay at chance. Second, models generally find artificial errors added to data easier to diagnose than recorded faults from the source systems, with this pattern holding in 32 of 44 model–dataset comparisons and reaching statistical significance in 18. Third, synthetic training improves cause identification on real time series and on recorded industrial faults. Finally, diagnosis accuracy alone does not determine recovery. Across 13 model, adapter, and control readouts, overall cause accuracy has Spearman correlation $\rho = -0.808$ with 60-step rollout error, while recall on changed-rule cases reaches $\rho = -0.946$. These findings show why revision should be evaluated from evidence, to diagnosis, to the update that follows, and finally to whether the system recovers.

Contributions. (i) A matched-triplet benchmark with an exact reference at every step, extended to real series and recorded faults. (ii) LOCUSPROBE: an 8B probe beats a 72B model's answers on identical items. (iii) PMT, which transfers from synthetic data to real faults without real labels, and PMT-Readout, which repairs the answer through the unembedding alone. (iv) A diagnosis-to-recovery evaluation in which PMT beats an exact-family observer in worst-case loss.

2 METHODS

2.1 IDENTIFYING THE CAUSE TELLS US WHAT TO REVISE

To distinguish possible causes of an error, we separate the system's actual state, the rule governing how it changes, and the observations reported to us. Let x_t denote the actual state at time t and a_t an action. The next state is

$$x_{t+1} = f_\theta(x_t, a_t), \tag{1}$$

where f_θ describes how the action changes the state and θ specifies the rule's parameters. We observe a reported value $\hat{x}_{t+1}$, which may differ from the actual state. For a robot, these quantities may describe its previous position x_t, a movement command a_t , the rule $f_\theta(\cdot)$ determining its movement, and the current position $\hat{x}_{t+1}$ reported by its sensor.

An *observation fault* makes the reported value incorrect while the system itself continues normally. A *state disturbance* changes the actual state x_t, but subsequent actions still follow the original rule. A *rule change* alters $f_\theta(\cdot)$, so the same action can now have a different effect. Identifying the cause of the fault therefore informs us whether to correct the observation, revise our estimate of the current state, or update the rule used to predict what happens next.

As an example, Table 1 shows an arithmetic triplet. A hidden counter follows $z \leftarrow z + 2$, and the model has seen $0, 2, 4, 6$, so it expects 8, but instead it sees 11. Up to this point the three examples are the same prompt, byte for byte, so the exact posterior is $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, and no answer can do better than chance. The next readings then diverge. If the reading was wrong (observation fault), the counter returns to its path (10, 12); if the state was knocked to 11 (state disturbance), it keeps counting by two from there (13, 15); if the rule changed to $z \leftarrow z + 5$ (rule change), it keeps adding five (16, 21). Here one further reading separates the three, and each calls for a different revision: discard the reading, reset the state, or replace the rule.

At step j, let H_j contain all observations available so far, and let c denote one of the possible causes. Because the benchmark specifies how each cause produces observations, we can calculate

Table 1: An example arithmetic triplet. The three sequences are identical through the anomalous reading ($t = 4$) and separate afterwards.

Cause	t=0	1	2	3	4	5	6
Observation fault ($11 \leftarrow 8$)	0	2	4	6	**11**	10	12
State disturbance ($11 \leftarrow 8$)	0	2	4	6	**11**	13	15
Rule change ($z \leftarrow z + 5$)	0	2	4	6	**11**	16	21
Exact posterior	uniform through t=4					point mass	

the probability of each cause given the available evidence:

$$P^\star(c \mid H_j) = \frac{P(H_j \mid c)P(c)}{\sum_{c'} P(H_j \mid c')P(c')}. \tag{2}$$

Here, $P(c)$ is the prior probability of a cause, and $P(H_j \mid c)$ is the likelihood of the observed history under that cause. This calculation provides an exact reference distribution for the causes and assumptions specified by the benchmark.

We create each example in three versions: one where the observation is wrong, one where the state has changed, and one where the rule has changed. Initially, the model sees the same history and the same unexpected observation in all three versions. The true cause differs, but nothing shown to the model reveals that difference yet. Since each cause occurs equally often, the appropriate prediction is to give each cause probability $P^\star(c \mid H_j) = 1/3$. A model that always selects the same cause will still get one of the three versions right. Its accuracy will therefore be $1/3$, even if it is completely confident in that choice.

To distinguish an unsupported confident guess from appropriate uncertainty, we check the probabilities assigned to all three causes as well as the selected answer. Further observations rule out the alternatives until only the cause used to generate the example remains possible. As evidence arrives, the model should update its probabilities and become confident once the evidence identifies one cause. It can fail either by committing to a cause before the evidence distinguishes it, or by remaining uncertain after the cause becomes clear.

2.2 DATASETS

The controlled benchmark has 200 triplets in each of three domains, arithmetic, logic, and navigation, for 600 triplets and 1,800 examples. Each triplet holds one example per cause. The three share a byte-identical history up to and including the anomalous reading, so the exact posterior there is uniform; two further observations then separate them, so the model is scored at three evidence stages, $j = 0, 1, 2$ (Appendix A). The same construction is planted on windows of real measurements: daily temperature from NOAA weather stations, server metrics from NAB, and the water-circulation testbed SKAB. A fault of 4 to 6 residual standard deviations is added as a one-reading spike (OBS), a level step (STATE), or a growing ramp (LAW). (Appendix B).

We use 303 recorded fault events from four testbeds (Katser & Kozitsin, 2020; Jacob et al., 2021; Shin et al., 2021; Keipour et al., 2019) (Section 3). Each event is labelled by a numerical reference on the real stream, giving 159 state disturbances and 144 rule changes. 400 episodes on real NOAA backgrounds cover five fault settings, 80 each; each diagnosis selects a revision (ignore, re-initialize the state, update the rule), and the forecast is scored against the true latent value up to 40 steps ahead.

2.3 LOCUSPROBE AND POSTERIOR-MATCHED TUNING

To read what a model knows, LOCUSPROBE is a linear classifier on its hidden state: one-vs-rest logistic regression on the last-prompt-token representation at the resolved evidence step, with the layer chosen by cross-validation on synthetic data only and folds grouped by triplet so no triplet straddles training and test. To make the model's own answer use that information, PMT trains a rank-16 LoRA adapter so that the model's distribution over the three answer letters matches the exact posterior at every evidence step:

$$\mathcal{L}_{\text{PMT}} = -\sum_j \sum_c P^\star(c \mid H_j) \log P_M(c \mid H_j). \tag{3}$$

Training uses 1,200 synthetic arithmetic and logic examples under all three option orders; navigation and all real series are held out. At the ambiguous step the target is uniform, so the model is trained to stay uncertain exactly where the evidence licenses nothing. PMT-Readout is the same objective with every weight frozen except the unembedding, so nothing in the network's computation can change and only the map from the final representation to the answer letters is learned (Appendix C).

2.4 EVALUATION AND CONTROLS

For every example, we ask the model to choose one of three answer categories and record the probabilities it assigns to them. We also change the order of the answer choices and average the results, so

that the model cannot succeed simply by favoring a particular position. The model answers immediately after reading the prompt, without an additional reasoning step. Further details are provided in Appendix A.

We evaluate three stages separately. First, we ask what a model reports as the cause and what a separately trained classifier can recover from its hidden states. Second, we ask whether that diagnosis leads to a useful update or action. These measurements must remain separate because a model may contain information about the cause without reporting it, or may report the correct cause but still choose the wrong action. We also verify that the hidden-state results are not caused by differences between otherwise identical inputs. Before later observations distinguish the causes, all three examples in each triplet are identical, so a classifier should perform no better than chance: an AUROC of 0.500 and three-way accuracy of $1/3$.

We report $\alpha =$ *probability agreement* as an evaluation metric, not as a training loss. Let $P_M(c \mid H_j)$ be the probability that the model assigns to cause c given the available history H_j, and let $P^\star(c \mid H_j)$ be the ground truth reference probability specified by the benchmark. We measure how closely these distributions agree using

$$\alpha = 1 - \frac{1}{2} \sum_c |P_M(c \mid H_j) - P^\star(c \mid H_j)| \,. \tag{4}$$

The value of α ranges from 0 to 1, with 1 indicating an exact match between the model and the reference distribution. When the evidence identifies one cause, the reference is a point mass, such as $P^\star(\cdot \mid H_j) = (0, 0, 1)$, and the score equals the probability assigned to the true cause. When the evidence is ambiguous, the reference is $P^\star(\cdot \mid H_j) = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, so the score rewards the model for expressing the correct uncertainty rather than confidently selecting one unsupported cause.

3 Experiments

The experiments follow the path from evidence to revision. We first measure how models answer on the synthetic benchmark and what their hidden states contain, then test the same comparison on real series and across the shift from synthetic to real data. We next compare recorded faults with injected ones, evaluate posterior-matched tuning and LocusProbe on recorded faults, and finally measure how much each diagnosis improves the forecast it selects.

Direct answers on the synthetic benchmark. One control shows directly what the answer channel does before any intervention. We present compound faults, an off-taxonomy slow drift, and inputs with no fault at all, and read the same three-way answer ($n = 120$ per cell, Table 13 in Appendix L). Qwen3-4B answers LAW in every cell, including when nothing happened, and Qwen2.5-7B answers STATE in 0.88–0.99 of each cell, both confidently, with mean entropy of 0.41–0.59 and 0.06–0.14 against $\log 3 \approx 1.099$. A model that names a rule change on healthy data at entropy 0.41 is not diagnosing but reporting a prior through a channel the evidence does not reach, and which constant a model reports depends on the condition set: on the synthetic benchmark Qwen3-4B names an observation fault in about nine of ten resolved examples (Appendix K). We then evaluate Qwen3-4B on the synthetic tasks at three evidence stages. At the initial stage the evidence cannot distinguish the three causes, and the model's accuracy is 0.332, close to the expected $1/3$. Additional observations make the cause increasingly identifiable, since a reference predictor that uses the benchmark's known rules achieves 0.889 at the intermediate stage and 1.000 at the final stage, yet the model's direct-answer accuracy remains at 0.336 and 0.327. The evidence therefore becomes sufficient to identify the cause, but the model does not use it in its direct answers. Table 14 in Appendix L shows that this pattern is not specific to Qwen3-4B: as the evidence arrives, the entropy of the reference distribution falls from $\log 3 \approx 1.099$ to 0.157 and then to 0, while the models' belief entropy stays nearly flat, the best of them reaches 0.459, and scale does not remove the pattern within the tested range.

Hidden-state probe on the synthetic benchmark. We then ask whether the cause can be recovered from the model's hidden states after the final observations. LocusProbe achieves 0.872 accuracy on held-out examples, compared with the model's direct-answer accuracy of 0.327, and the same probe reaches 0.765 for Qwen3-1.7B and 0.754 for R1-Distill-1.5B. Under the same folds,

a classifier on the embedding layer is at chance and logistic regression on surface tokens reaches 0.474–0.522, so the hidden states contain information useful for distinguishing the causes even when the tested answer procedure performs poorly (Appendix N). To test whether the model uses this representation, we swap it: the resolved last-token hidden state of a LAW example is written into its matched OBS example at one layer, and we measure how $P(\text{LAW})$ moves. In Qwen2.5-7B the swap raises $P(\text{LAW})$ by $+0.116$ to $+0.120$ at layers 20–27 and by at most $+0.012$ below layer 16, while a same-cause donor moves it by at most 0.03 and a pre-resolution donor pushes it the other way, so the cause is carried by the upper layers once the evidence has arrived.

Probes and direct answers on real series. Table 6 in Appendix E compares the aligned probes with direct answers. On NOAA-based items, Llama-3.1-8B reaches 0.973 three-way probe accuracy against 0.438 direct-answer accuracy on the same items. On the 120 items where Qwen2.5-72B's direct answers exist, the Llama-3.1-8B probe reaches 0.967 against the 72B model's 0.842, a margin of $+0.125$ $[+0.080, +0.173]$ from a model nine times smaller, with 16 items the probe alone gets right against 1 the other way (McNemar $p = 0.00027$). More broadly, the probe beats all eight tested open direct-answer systems on matched NOAA items, with every paired 95% interval above zero, and reaches 0.900 accuracy with only 25 target labels. The gap also persists at larger scale: at 32B, the probe reaches 0.977 on NOAA and 0.870 on SKAB, while that model's direct answers reach 0.54–0.75, and direct answers remain far below the probe under every elicitation we tried (Appendix M). The same holds against a time-series foundation model, since under matched folds the probe leads MOMENT-1-large with its own classification head by .335 [.306, .365], and two independent fits of that head give .642 and .652. GPT-5, which does not expose token probabilities and is scored by committed answers, is evaluated separately and also falls below the probe (Appendix F).

Transfer from synthetic to real series. The preceding probes are trained within the same domain in which they are tested. We next ask whether the representation transfers across domains, training a linear probe on hidden states from synthetic arithmetic and logic examples only, selecting its layer on synthetic folds only, and freezing the readout before evaluating it on real time series, so that no target-domain labels are used. For Llama-3.1-8B, the resulting layer-16 probe is above the three-way chance level on every real domain, from 0.528 on NOAA under the strict accounting to 0.739 raw accuracy on the 46 recorded SKAB faults (Appendix E). Classifiers on the input itself, trained on the same synthetic data, stay at chance on NOAA, NAB, and anchored SKAB, so the transferable cause information is computed by the pretrained network rather than read off the input, although the effect is model-dependent, with Qwen2.5-7B transferring less consistently.

Recorded versus injected faults. The recorded-fault evaluation covers four sources: a water-circulation testbed (SKAB), Spark cluster traces (Exathlon), an industrial-control testbed (HAI 21.03), and flight-failure recordings (ALFA). For each source, the study compares recorded anomalies with matched injections based on the same logs, and class-standardized comparisons and clustered uncertainty address the dependence between matched cases and differences in cause frequencies. Across 44 model–source cells, injection-minus-recorded gaps are positive in 32 and significant in 18, with at least one significant cell in each source (Table 10 in Appendix H). Pooling 7,034 recorded-versus-injected items over 11 models and the four testbeds, with model, source, and class fixed effects and event-clustered errors, injected anomalies overstate accuracy on recorded faults by $+0.135$ $[0.094, 0.176]$, and by $+0.166$ $[+0.010, +0.320]$ on the nine cells with shared STATE/LAW support. Residual-preserving twins identify what the matching leaves out. These injected cases reuse the recorded post-onset residuals, so both arms carry the same real continuation noise and differ only in the planted cause, and under this construction the advantage is no longer detected: the mean gap over the complete 11×4 grid is -0.034 $[-0.078, +0.008]$ (4,000 cluster-bootstrap resamples), with no confidence gate applied to either arm. No cell is significant after Holm correction, and no model at or above 7B retains a positive gap on SKAB, even when the twins are restricted to reference-confirmed labels. Anomalies injected with clean templates therefore overstate transfer to recorded faults, and residual-preserving twins provide a stricter test (Appendix H).

Posterior-matched tuning. Synthetic supervision also improves the model's own answers. On 1,800 held-out examples PMT raises Qwen2.5-7B's intermediate posterior agreement from .443 to .851 and resolved accuracy from .461 to .874, exceeding matched hard-label tuning (Appendix J). Synthetic-only PMT transfers to NOAA and SKAB, reaching .597 and .580 for Qwen2.5-7B and

Table 2: Recorded-fault transfer. STATE/LAW balanced accuracy on 303 events. The last column gives event-clustered 95% intervals for the pooled value, where available. Model rows use restricted token probabilities. The best value in each column is in bold and the second best is underlined.

System	SKAB (46)	Exathlon (107)	HAI (85)	ALFA (65)	Pooled (303)	95% CI
Constant class	.500	.500	.500	.500	.500	–
Raw-reading logistic regression	.394	.647	.603	.543	.534	–
IMM, noise fitted on held-out NOAA	.500	.603	.619	.847	.628	–
MOMENT-1-large	<u>.774</u>	.566	.726	.611	.665	–
ChatTS-14B	.567	.532	.809	.690	.681	[.64, .73]
Qwen2.5-32B	.755	.672	.817	.745	.737	[.70, .78]
gpt-4.1-mini	.702	.678	<u>.845</u>	<u>.853</u>	<u>.779</u>	[.74, .82]
phi-4	.700	<u>.702</u>	.793	.678	.736	[.70, .77]
phi-4 + PMT	.640	.657	.826	.765	.763	[.73, .80]
Llama-3.1-8B	.677	.605	.707	.481	.628	[.59, .66]
Llama-3.1-8B + LOCUSPROBE	**.867**	**.885**	**.862**	**.860**	**.865**	[.83, .90]

.730 and .630 for Qwen3-8B, and both models also improve on residual-preserving SKAB faults (.384 $\rightarrow$ .543 and .326 $\rightarrow$ .601). Where the change happens inside the model points to the answer channel: an explicit rule-change note yields zero LAW recall, whereas training only the output mapping raises recall from .025 to .721 $\pm$ .007, and upper-half LoRA reaches .623 $\pm$ .019 revision accuracy against .532 for all layers and .468 for the lower half, matching the layers where the representation swap acts.

Diagnosis on recorded faults. Trained on constructed NOAA and NAB faults, LOCUSPROBE leads every system on all four recorded-fault testbeds (Table 2), with no recorded labels or test series entering training. Pooled balanced accuracy reaches .865, exceeding gpt-4.1-mini by .086 [.042, .136], MOMENT-1-large by .200 [.156, .245], and an interacting-multiple-model filter fitted on held-out NOAA by .237, while the same classifier on raw readings reaches .534. Trained on synthetic data alone it still leads TabPFN v2, Toto 2.0 and MOMENT, and no testbed appears in any baseline's documented training data (Appendix H).

Revision and forecast recovery. Each diagnosis selects a fixed revision, discarding the reading for OBS, resetting the level for STATE, or updating the drift for LAW, and all systems use the same revision operators. Forecast loss is $|\hat{x} - x|/\sigma$, with $\hat{x}$ the forecast, x the true latent value, and σ the background residual scale, and E is its mean over the 60-step evaluation. Recovery $R = (E_0 - E)/(E_0 - E_\star)$ is the share of the oracle's error reduction achieved, with $E_0 = 106.26$ the error without revision and $E_\star = 11.19$ the error when the true cause selects the revision, so recovery measures the consequence of a diagnosis rather than its accuracy.

Figure 2 shows that probe-selected revisions recover 98.5–99.9% of the oracle's improvement (full values in Table 4). For Qwen3-8B, replacing the direct answer with the probe reduces error from 105.59 to 12.39, raising recovery from 0.7% to 98.7%, while MOMENT-1-large, a time-series foundation model selecting the same revisions, recovers 5.4% and three forecasting foundation models given the same record and no diagnosis recover 11.7–13.9%. On the separate 400-episode benchmark at $h = 40$, Qwen2.5-7B with the probe achieves the lowest reported worst-case loss (6.63) and harmful-revision rate (17.0%) among measured revision systems, and the Llama-3.1-8B probe keeps its margin over an observer given the exact generative family at every horizon, from -1.02 $[-1.39, -0.65]$ at $h = 1$ to -8.34 $[-10.18, -5.84]$ at $h = 40$, with PMT also leading the observer at $h = 40$ (-3.83 $[-6.28, -0.56]$). Missed rule changes compound across future predictions, and across the original 13 readouts LAW recall correlates more strongly with lower forecast error than overall diagnosis accuracy ($\rho = -.946$ versus $-.808$). The association is specific to the compounding cause, since STATE recall reaches only $\rho = -.396$ ($p = .18$), a difference in magnitude of $+.550$ $[+.074, +.935]$.

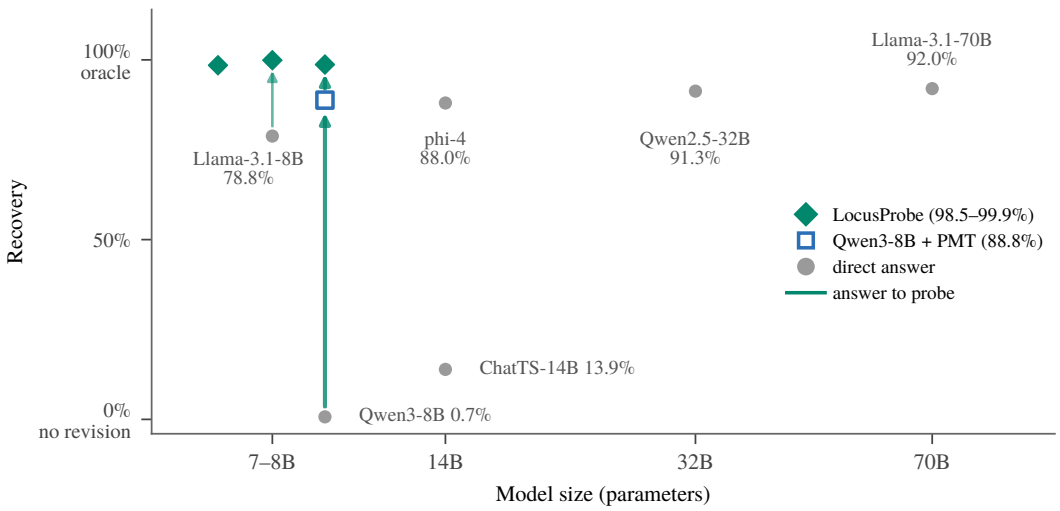


Figure 2: Recovery of the oracle's error reduction over a 60-step NOAA forecast, by model size. All systems choose among the same three revision operators. Arrows join a model's direct answer to LOCUSPROBE on the same model, and for Qwen3-8B the arrow passes through PMT; the three 7–8B models are offset horizontally for legibility. MOMENT-1-large, which selects revisions through its own classification head, recovers 5.4%, and native forecasters, which do not choose a revision, recover 11.7–13.9% (Table 4).

4 RELATED WORK

Deciding whether a discrepancy comes from the sensor, the state, or the dynamics is the classical problem of fault detection and isolation (Willsky, 1976; Isermann, 2006). Likelihood-ratio tests, multiple-model filters, and Bayesian changepoint inference answer it exactly when the model class is right (Willsky & Jones, 1976; Blom & Bar-Shalom, 1988; Adams & MacKay, 2007), which is why we use an exact-family observer and an interacting-multiple-model filter as baselines. Model-based diagnosis in artificial intelligence treats the same problem symbolically, identifying the smallest set of components whose failure explains an observed inconsistency (Reiter, 1987; de Kleer & Williams, 1987). Our three causes can be read as three such candidate components of a world model, and the matched triplets make the evidence that separates them explicit. Learned models approach the same data from the forecasting side. Language models have been used as zero-shot forecasters (Gruver et al., 2023), reprogrammed for forecasting (Jin et al., 2024), applied to anomaly detection (Alnegheimish et al., 2024), and aligned to numeric series with synthetic data (Xie et al., 2025), and specialist models are pretrained as forecasters or as encoders with task heads (Goswami et al., 2024; Ansari et al., 2024; Das et al., 2024; Hollmann et al., 2025). These methods ask whether something is wrong or what comes next (Chandola et al., 2009). We ask which part of a model a surprise should change, and evaluate all of them on that question under one protocol.

The question also differs from belief revision in language models. Belief-change theory and truth maintenance formalize which commitments to retract when information arrives (Alchourrón et al., 1985; Doyle, 1979), and empirical work tests whether models update when a premise changes and whether edits reach their consequences (Longpre et al., 2021; Xie et al., 2024; Wu et al., 2024; Cohen et al., 2024). Work on knowledge conflicts studies how models weigh evidence in context against what they learned in training (Xu et al., 2024). In all of this work the changed premise is named, whereas here the evidence only shows that something is wrong and the model has to infer which part of its account to change. To test whether models nonetheless represent the cause, we draw on probing. Linear probes measure what a representation makes accessible and need controls to be interpreted (Alain & Bengio, 2017; Hewitt & Liang, 2019; Belinkov, 2022), latent-knowledge studies find internal signals that outrun stated answers (Burns et al., 2023; Orgad et al., 2025; Kadavath et al., 2022), and activation patching tests whether a decoded quantity affects behavior (Vig et al., 2020; Geiger et al., 2021). Since chains of thought need not reflect the computation behind an

answer (Turpin et al., 2023; Lanham et al., 2023), we compare internal readouts with stated answers directly.

Training on synthetic data and testing on real data follows a long line of work on transfer from simulation, where varied simulated conditions are used to reach real ones (Tobin et al., 2017); our synthetic worlds play that role for diagnosis, and the recorded faults test whether the transfer holds. Diagnosis at the ambiguous stage also relates to selective prediction, in which a model abstains when it cannot answer reliably (Geifman & El-Yaniv, 2017; Kamath et al., 2020). Our exact posterior tells us when abstention is warranted, and the probability-agreement score rewards uncertainty at exactly those points. Finally, anomaly benchmarks often inject faults into recorded series, a practice whose validity has been questioned (Wu & Keogh, 2023). Our recorded-versus-injected comparison measures how much this inflates results, and the matched-triplet design supplies an exact posterior, so uncertainty can be scored against the evidence and not only calibrated (Guo et al., 2017). Diagnosis in this paper assumes a detected discrepancy. Recorded-fault labels come from a numerical reference and cover only STATE and LAW, probe success shows external accessibility rather than unaided use, recovery uses fixed revision operators, and controlled, injected, and recorded faults remain distinct evaluations.

5 Conclusion

Diagnosing a prediction error means deciding whether the observation, the current state, or the governing rule should be revised, and our results show that language models often hold the information needed for that decision without expressing it in their answers. LocusProbe reads this information from the hidden state with .973 accuracy on matched NOAA items, where the same model's direct answers reach only .438, and it reaches .865 pooled balanced accuracy on recorded faults from four testbeds even though no recorded label is used in training. Because the diagnosis selects the revision, these gains carry through to forecasting: replacing Qwen3-8B's direct answer with its probe raises oracle-relative recovery from 0.7% to 98.7%, and probes on 7–8B models recover more of the oracle's improvement than the direct answers of models up to nine times larger.

These findings suggest two changes to how diagnosis is evaluated. The first is to judge a diagnosis by what it does to later predictions rather than by overall accuracy alone, since an uncorrected rule change keeps compounding while an uncorrected observation error fades, which is why recall on rule changes predicts recovery far better than accuracy does. The second is to test on recorded failures and not only on injected ones, because across four testbeds models score higher on injected anomalies than on recorded faults from the same logs, and that advantage is no longer detected once the injected cases keep the recorded residuals, so benchmarks for diagnosis are best built from recorded faults or residual-preserving twins alongside controlled examples with exact references.

Several directions follow naturally from this work. Diagnosis here begins once a discrepancy has been detected, and joining detection and diagnosis in a single model would bring the setting closer to deployment, while extending the taxonomy beyond three single causes to simultaneous and unfamiliar faults would show how far the learned representations generalize. The revision operators are also fixed in advance, so learning them together with the diagnosis could recover more than any operator chosen by hand, and closed models that expose token probabilities could then be compared on exactly the same footing as the open models studied here. More broadly, the distance between what models represent and what they report suggests that part of the value of a language model as a diagnostician lies in reading its internal states directly, and PMT shows that a meaningful share of that value can be brought back into the model's own answers through training on synthetic examples alone.

Reproducibility statement. Code, generators, prompts, and configurations for every experiment are available in an anonymized repository at `https://anonymous.4open.science/r/locus-revise`. The appendices describe the controlled studies, the separate cohorts, and the response procedures.

AI use statement. Generative AI tools were used to help draft and edit the text of the manuscript.

REFERENCES

Ryan Prescott Adams and David J.C. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (Workshop Track)*, 2017. URL https://arxiv.org/abs/1610.01644.

Carlos E Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2), 1985.

Sarah Alnegheimish, Linh Nguyen, Laure Berti-Equille, and Kalyan Veeramachaneni. Large language models can be zero-shot anomaly detectors for time series? In *IEEE International Conference on Data Science and Advanced Analytics*, 2024.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022. doi: 10.1162/coli_a_00422. URL https://aclanthology.org/2022.cl-1.7/.

Henk A. P. Blom and Yaakov Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, 1988. doi: 10.1109/9.1299.

Craig Boutilier. A unified model of qualitative belief change: A dynamical systems perspective. *Artificial Intelligence*, 98(1–2):281–316, 1998. doi: 10.1016/S0004-3702(97)00066-0. URL https://doi.org/10.1016/S0004-3702(97)00066-0.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–58, 2009.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 2024. doi: 10.1162/tacl_a_00644. URL https://aclanthology.org/2024.tacl-1.16/.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning*, 2024.

Johan de Kleer and Brian C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 32(1): 97–130, 1987.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jon Doyle. A truth maintenance system. *Artificial Intelligence*, 12(3):231–272, 1979. doi: 10.1016/0004-3702(79)90008-0.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2106.02997.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *International Conference on Machine Learning*, 2024.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. Large language models are zero-shot time series forecasters. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025.

Rolf Isermann. *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer, 2006.

Vincent Jacob, Fei Song, Arnaud Stiegler, Bijan Rad, Yanlei Diao, and Nesime Tatbul. Exathlon: A benchmark for explainable anomaly detection over time series. *Proceedings of the VLDB Endowment*, 14(11):2613–2626, 2021. URL https://arxiv.org/abs/2010.05073.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, 2024.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In *Proceedings of ACL*, 2020.

Iurii D. Katser and Vyacheslav O. Kozitsin. Skoltech anomaly benchmark (SKAB). Kaggle, 2020. URL https://github.com/waico/SKAB.

Azarakhsh Keipour, Mohammadreza Mousaei, and Sebastian Scherer. ALFA: A dataset for UAV fault and anomaly detection. *arXiv preprint arXiv:1907.06268*, 2019. URL https://arxiv.org/abs/1907.06268.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Alexander Lavin and Subutai Ahmad. Evaluating real-time anomaly detection algorithms – the numenta anomaly benchmark. *arXiv preprint arXiv:1510.03336*, 2015. URL https://arxiv.org/abs/1510.03336.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roee Aharoni, Idan Szpektor, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *International Conference on Learning Representations (ICLR)*, 2025.

Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.

Hyeok-Ki Shin, Woomyo Lee, Jeong-Han Yun, and Byung-Gi Min. Two ICS security datasets and anomaly detection contest on the HIL-based augmented ICS testbed. In *Cyber Security Experimentation and Test Workshop*, pp. 36–40, 2021. doi: 10.1145/3474718.3474719. URL https://doi.org/10.1145/3474718.3474719.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2305.04388.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, 2020.

Bryan Wilie, Samuel Cahyawijaya, Etsuko Ishii, Junxian He, and Pascale Fung. Belief revision: The adaptability of large language models reasoning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10480–10496. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.586. URL https://aclanthology.org/2024.emnlp-main.586/.

Alan S. Willsky. A survey of design methods for failure detection in dynamic systems. *Automatica*, 12(6):601–611, 1976.

Alan S. Willsky and Harold L. Jones. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control*, 21(1):108–112, 1976. doi: 10.1109/TAC.1976.1101146.

Renjie Wu and Eamonn J. Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *IEEE Transactions on Knowledge and Data Engineering*, 35(3):2421–2429, 2023. doi: 10.1109/TKDE.2021.3112126. URL https://arxiv.org/abs/2009.13807.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2024.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *International Conference on Learning Representations (ICLR)*, 2024.

Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. ChatTS: Aligning time series with LLMs via synthetic data for enhanced understanding and reasoning. *arXiv preprint arXiv:2412.03104*, 2025. URL https://arxiv.org/abs/2412.03104.

Haoming Xu, Weihong Xu, Zongrui Li, Mengru Wang, Yunzhi Yao, Chiyu Wu, Jin Shang, Yu Gong, and Shumin Deng. When should models change their minds? contextual belief management in large language models. *arXiv preprint arXiv:2605.30219*, 2026. URL https://arxiv.org/abs/2605.30219.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. In *Proceedings of EMNLP*, 2024.

A CONTROLLED-STUDY METHODS

We begin with simulated systems in which we control what happens and know the cause of every error. We then use examples based on real measurements collected over time from NOAA, NAB, and SKAB. These examples test whether models can identify causes beyond the simple simulated tasks. Next, we ask whether models find artificial errors easier to diagnose than failures that occurred in real data. Using SKAB, Exathlon, HAI 21.03, and ALFA, we compare two types of examples: failures recorded while a system was running, and artificial errors created by changing values in its recorded data (Katser & Kozitsin, 2020; Lavin & Ahmad, 2015; Jacob et al., 2021; Shin et al., 2021; Keipour et al., 2019). Both use real data, but only the first captures the system's response to a failure. While some of these failures were deliberately induced in the physical system, they nevertheless still differ from artificial changes to recorded data, as the failure occurred in a real system and its downstream effects were measured. Throughout the paper, we use *recorded faults* to distinguish these recorded events from faults injected into real data streams.

A.1 DOMAINS

In the arithmetic domain, a counter follows $z \leftarrow az + b$. An anomalous reading can be a reporting error, a displacement of z, or a new pair (a', b') that yields the same initial anomaly. Enumerating the allowed disturbances and rules gives the exact posterior.

In the logic domain, a switch controls a lamp. An unexpected off reading when the switch is up can indicate a misreading, a burnt-out bulb, or inverted wiring. A down-switch observation distinguishes inverted wiring, and a subsequent up-switch observation distinguishes a transient misreading from a burnt-out bulb.

In navigation, a robot moves through a corridor. An unexpected reported position can arise from a sensor fault, displacement, or a changed movement gain. Later commanded movements distinguish the causes.

We generate 200 triplets per domain: 600 triplets and 1,800 instances in total. The three members share their initial history, so the triplet is the unit used for data splitting and bootstrap uncertainty.

A.2 MODEL CONFIGURATIONS AND ACTIVATION CAPTURE

The main model configurations include Qwen3-4B, Qwen3-1.7B, DeepSeek-R1-Distill-Qwen-1.5B, and Qwen2.5-3B/7B (Team, 2025; 2024; DeepSeek-AI, 2025). The forced-answer readout bypasses the thinking block, so both Qwen3-1.7B modes share one computation and are reported once.

Hidden-state classifiers use triplet-grouped folds, with layer selection on inner training folds. Deterministic activation capture uses batch size one and disables TF32. At the ambiguous step, identical triplet prompts produce identical activations, providing an exact $1/3$ reference for classification.

B IMPLEMENTATION DETAILS

Real-series triplets. A background window is drawn from a real series with a pre-window of 12, 16, or 20 readings and 4 evidence readings. A least-squares local linear trend is fitted to the pre-window, and σ is the RMS of its residual; flat windows are rejected. The fault magnitude is $\Delta = s\sigma$ with $s \sim U(4, 6)$ and a random sign. Writing m for the index after onset, the three worlds add Δ at $m = 0$ only (OBS), Δ at every m (STATE), and $\Delta(m + 1)$ (LAW), so they are byte-identical through the onset reading; a triplet is discarded unless the reference posterior's first entries agree to 10^{-6}. A window is also discarded if its own real continuation jumps by more than $3\sigma + 0.5|\Delta|$, so a natural jump cannot be confounded with the planted cause.

Numerical reference. Three pre-window models are stored on every item: a local linear trend, an AR(1) on the trend residual, and a random-walk local level. For each cause the magnitude is integrated over a fixed grid of ± 2 to $\pm 12\sigma$ under a Gaussian likelihood with uniform class priors, giving the exact posterior at every evidence step.

Prompts and readout. Each prompt lists the record so far and the three causes as lettered options, under all three cyclic rotations, and ends in an assistant prefill inside `\boxed{`. The readout is the softmax of the final-position logits restricted to the option letters; the letters carry at least 0.9996 of the next-token mass in every run (0.99995 on the NOAA run). Results are re-indexed to canonical order and averaged over rotations.

Probes. Features are the last-prompt-token hidden state at the resolved evidence step. The classifier is one-vs-rest logistic regression ($C = 0.05$, liblinear) on standardized features, with four triplet-grouped folds and the layer chosen by inner-fold cross-validation on the synthetic training domains only. The two transfer accountings, *strict* (training scaler) and *re-standardized* (target feature statistics, no labels), are never mixed within a comparison.

Revision benchmark. Five settings of 80 episodes each ($N_{\mathrm{pre}} = 16$, 4 evidence readings, horizon 40). The three actions are computed from the observed record alone: ignore follows the pre-window trend, state re-initialization adds the mean post-onset residual, and a rule update adds the least-squares drift. The argmax rule maps the modal cause to its action; the expected-cost rule minimizes expected loss under the diagnosis. Loss is $|\mathrm{forecast} - \mathrm{truth}|/\sigma$ against the true latent value, and a harmful revision is an action worse at $h = 40$ than not revising.

Statistics and seeds. Episode bootstraps use $4{,}000$–$8{,}000$ resamples with the same indices across arms, so contrasts are paired; worst-case bootstraps are stratified within setting. Recorded-fault intervals resample the 166 event clusters. Cell-wise tests are corrected with Holm and Benjamini–Hochberg. The central adaptation runs use three optimisation seeds.

C Adapter configuration

Readout. Each prompt ends inside a `\boxed{` prefill, and we read the next-token probabilities of the three option letters in a single forward pass and renormalize over them; the letters carry 0.9996–1.0000 of the next-token probability mass. Results are averaged over the three cyclic rotations of the options.

Training. The adapter is trained with the cross-entropy between this three-letter readout and the exact posterior $P^{\star}(\cdot \mid H_j)$ at every evidence step, so ambiguous steps receive soft targets. LoRA uses rank 16, scaling 32, dropout 0.05, and all attention and MLP projections. Learning rate is 10^{-4} with cosine decay and effective batch size 16. Training runs for two epochs over $3{,}600$ rendered prompts from $1{,}200$ new instances, with generator seeds disjoint from evaluation. Computation uses bf16 on a single 24 GB GPU; the reported Qwen3-1.7B run requires approximately fifteen GPU-minutes.

D Identity and splitting checks

At the anomaly, every matched triplet has byte-identical prompts under each answer rotation. Deterministic re-execution verifies zero within-triplet activation differences. A classifier must therefore give the same prediction to all three members and obtains exactly one correct answer per triplet. All downstream probe folds and bootstrap intervals preserve triplet grouping.

The three examples in a triplet share the same initial history, so we treat the entire triplet as one unit when splitting the data and estimating uncertainty. This prevents a decoder from training on one version of a triplet and being tested on another nearly identical version. We choose the hidden layer using only the training portion of each fold, rather than using the final test labels to select the most favorable layer. At the beginning of each triplet, all three examples show the model exactly the same information, even though they were generated from different causes. The model must therefore give the same answer to all three examples. That answer can match only one of the three causes, so the correct accuracy at this stage is exactly $1/3$.

For the real-series probes, we use four-fold cross-validation while keeping all three members of each triplet in the same fold. This prevents the probe from training on one version of a triplet and being

tested on another. Within each training fold, we select the hidden layer using only the training data, and we shuffle the cause labels as a permutation control.

E REAL-SERIES AND RECORDED-FAULT RESULTS

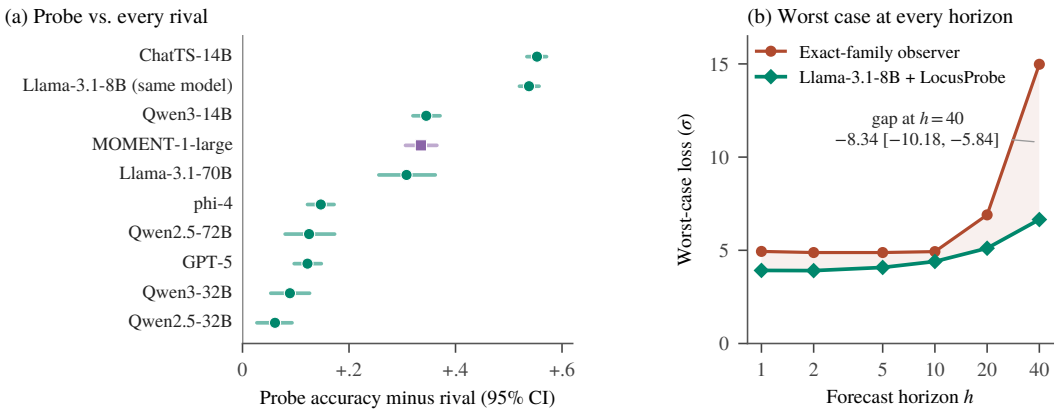


Figure 3: (a) Probe accuracy minus each rival's accuracy on matched NOAA items, with paired 95% intervals (Table 3). (b) Worst-case forecast loss across the five revision settings as the horizon grows, for the Llama-3.1-8B probe and an observer given the exact generative family of all three causes.

Table 3: Values plotted in Figure 3a: probe accuracy minus each rival's accuracy on matched NOAA items, with paired 95% intervals.

Rival	Margin [95% CI]
ChatTS-14B	+.553 [+.534, +.571]
Llama-3.1-8B, same backbone	+.538 [+.520, +.557]
Qwen3-14B	+.345 [+.320, +.371]
MOMENT-1-large (own head)	+.335 [+.306, +.365]
Llama-3.1-70B	+.308 [+.256, +.362]
phi-4	+.147 [+.122, +.172]
Qwen2.5-72B	+.125 [+.080, +.173]
GPT-5	+.122 [+.097, +.148]
Qwen3-32B	+.089 [+.053, +.126]
Qwen2.5-32B	+.061 [+.027, +.093]

On the aligned NOAA items the Llama-3.1-8B probe reaches 0.996 AUROC alongside its 0.973 accuracy, and the same capture applied to a randomly initialized network of the same architecture reaches only 0.533, so pretraining supplies most of the recoverable information.

Under the strict accounting the layer-16 Llama-3.1-8B probe reaches 0.528 on NOAA, 0.668 on NAB, 0.544 on anchored SKAB, and 0.739 raw accuracy on the 46 recorded SKAB faults; re-standardizing on unlabeled target examples gives 0.755–0.768 on the first three and lowers the recorded-fault cell to 0.609. Qwen2.5-7B reaches 0.239 on that cell strictly, and 0.652, 0.632, and 0.703 on NOAA, NAB, and anchored SKAB after re-standardization. Input-level classifiers trained on the same synthetic data score 0.333 on NOAA, NAB, and anchored SKAB from raw readings and 0.333–0.347 from engineered residual features; on recorded SKAB, where the majority class gives 0.674, the best feature baseline reaches 0.565 against the probe's 0.739.

Robust where the observer is brittle. The observer assumes it knows how a perturbation propagates. Changing that operator, with the settings and actions fixed and the adapter still zero-shot, costs the observer up to 0.070 in accuracy, while the adapter moves by at most 0.035 in either direction. The adapted arms hold the best worst case at every horizon from $h = 10$ onward (Table 5).

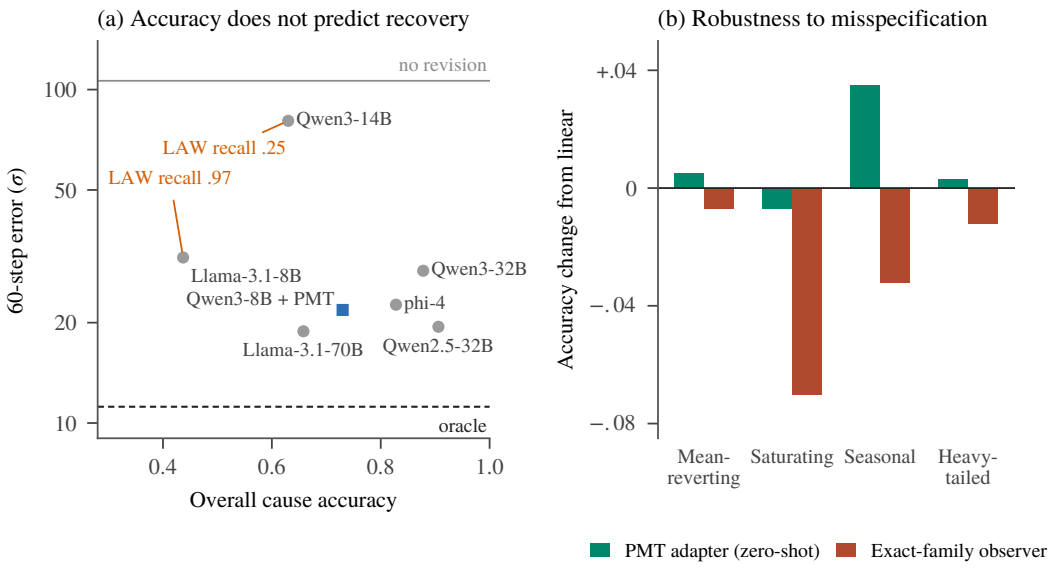


Figure 4: (a) Overall cause accuracy against 60-step error for the direct-answer readouts, on a log scale. Accuracy does not order recovery: Llama-3.1-8B, with low accuracy but high LAW recall, recovers better than Qwen3-14B. Qwen3-32B (28.62) and Qwen3-14B (80.6) come from the same rollout campaign. (b) Accuracy change from the linear propagation operator for the zero-shot PMT adapter and the exact-family observer.

Accuracy is not decision-equivalent. Let $C_{ij}(h)$ be the measured mean h-step loss of taking the action for cause j when the truth is i, $p(i)$ the class prior, and $P(\hat{c} = j \mid c = i)$ a policy's confusion. Its revision risk is $R_h = \sum_{ij} p(i)\, P(\hat{c} = j \mid c = i)\, C_{ij}(h)$. Across all policies, the rank correlation between accuracy and R_h is -0.059, $+0.081$, -0.176, and -0.372 at $h = 1, 5, 10$, and 40: accuracy is at most weakly related to the decision cost at any horizon ($|\rho| \leq 0.372$), so diagnosis should be scored by R_h at the horizon of use.

This appendix collects the real-series tables and Figure 4 referenced in Section 3: aligned probes against direct answers (Table 6), the 60-step rollout (Table 4a), worst-case loss across horizons (Table 5), attribution results, and the recorded-fault class composition.

Table 4: Forecasting outcomes of diagnosis-selected revisions. The 60-step evaluation reports mean normalized absolute error and oracle-relative recovery. The separate 40-step benchmark reports worst-case normalized loss and harmful-revision rate over 400 episodes. A revision is harmful when its error exceeds no revision on the same episode. Native forecasters use no diagnosis and no revision. Arrows indicate the preferred direction; NA denotes unmeasured results. The best diagnosis-driven value in each column is in bold and the second best is underlined; the reference rows are excluded.

	60-step evaluation		Separate 40-step benchmark	
System	Error ↓	Recovery ↑	Worst loss ↓	Harmful (%) ↓
Llama-3.1-8B + LOCUSPROBE	**11.32**	**99.9%**	<u>6.65</u>	20.3
Qwen3-8B + LOCUSPROBE	<u>12.39</u>	<u>98.7%</u>	7.59	<u>18.0</u>
Qwen2.5-7B + LOCUSPROBE	12.62	98.5%	**6.63**	**17.0**
Llama-3.1-70B	18.84	92.0%	NA	NA
Qwen2.5-32B	19.44	91.3%	NA	NA
Qwen3-8B + PMT	21.83	88.8%	11.15	22.8
phi-4	22.64	88.0%	NA	NA
Llama-3.1-8B	31.36	78.8%	NA	NA
GPT-5	48.52	60.7%	NA	NA
Raw-reading logistic regression	62.46	46.1%	NA	NA
ChatTS-14B	93.02	13.9%	22.49	32.3
MOMENT-1-large	101.10	5.4%	NA	NA
Qwen3-8B	105.59	0.7%	NA	NA
Native forecasters (no diagnosis, no revision)				
TimesFM 3.0	93.06	13.9%	NA	NA
Chronos-2	94.64	12.2%	NA	NA
Toto 2.0	95.11	11.7%	NA	NA
No revision	106.26	0.0%	26.81	0.0
Oracle revision (reference)	11.19	100.0%	NA	NA

Table 5: Worst-case loss across horizons (σ). The leading policy changes three times, and the adapted arms hold it from h=10 onward. Which cause dominates the worst case is horizon-dependent by construction: under *ignore* an uncorrected drift costs $|s|(h+1)$ while an uncorrected offset stays bounded, and the measured rule-change to state-shift cost ratio grows 1.04, 1.70, 2.96, 5.14 at $h = 5$, 10, 20, 40.

Policy	h=1	5	10	20	40
Always LAW	**2.78**	4.58	7.38	13.03	24.35
Always ignore	5.04	5.14	8.19	14.41	26.81
Base, expected-cost	3.21	**3.29**	5.91	11.98	24.30
Qwen2.5-7B adapter, expected-cost	3.27	3.41	**4.17**	9.18	19.48
Qwen3-8B adapter, expected-cost	3.07	3.64	4.69	**6.84**	**11.15**

Table 6: Aligned real-series probes and direct reports. Every listed ambiguous-stage cell is at AUROC 0.500 and three-way accuracy $1/3$. The random-init row applies the same capture to an untrained network with Llama-3.1-8B's architecture. Direct accuracy is on the 600 aligned items; the 300-item adaptation set gives 0.307 for Qwen2.5-7B.

Model	Source	Probe AUROC	Probe accuracy	Direct accuracy
Llama-3.1-8B	NOAA	.996	.973	.438
Qwen2.5-7B	NOAA	.990	.953	.300
Qwen3-8B	NAB	.985	.963	–
Random-init Llama-3.1-8B	NOAA	–	.533	–

Table 7: Real-series attribution results for Qwen2.5-7B. The adapters are trained only on synthetic arithmetic and logic examples.

Evaluation set	Base	Adapted
NOAA anchored	.307	.597
SKAB mixed	.428	.580
SKAB residual-preserving	.384	.543
Recorded SKAB	.261	.435
Recorded ALFA	.062	.385
Recorded Exathlon	.243	.477
Recorded HAI 21.03	.235	.376

Table 8: Class composition of the recorded-fault evaluation. Labels are operational (Appendix H); no OBS event survives the labelling filter.

Source	N	OBS	STATE	LAW
SKAB	46	0	31	15
ALFA	65	0	51	14
Exathlon	107	0	38	69
HAI 21.03	85	0	39	46
Total	303	0	159	144

F GPT-5 EVALUATION

GPT-5 and GPT-5-nano do not expose token probabilities, so they cannot be read out with the restricted-probability protocol used for every other model. They are scored with a separate committed-answer rule, and their results are reported here rather than beside the other models. Without token probabilities, GPT-5 systems are scored by majority STATE/LAW commitment across three rotations, with noncommitments counted incorrect. GPT-5 and GPT-5-nano cover 39.6% and 53.1% of events, scoring .875 and .600 when committing. The primary comparison uses gpt-4.1-mini under the shared probability-based protocol.

On matched NOAA items the Llama-3.1-8B probe exceeds GPT-5 by 0.122 $[0.097, 0.148]$, and with only 25 target labels by 0.058 $[0.031, 0.089]$. In the 60-step rollout, GPT-5's diagnoses recover 60.7% of the oracle's improvement (Table 4).

Table 9: GPT-5 on the recorded faults, STATE/LAW balanced accuracy under the committed-answer rule.

System	SKAB	Exathlon	HAI	ALFA	Pooled
GPT-5	.229	.353	.403	.436	.355
GPT-5-nano	.211	.483	.214	.193	.286

G INTERACTING-MULTIPLE-MODEL FILTER WITH REFERENCE-FAMILY NOISE

A second interacting-multiple-model variant uses the noise family of the numerical reference that labels the recorded faults, so it shares the labeller's modelling assumptions. Only its pooled result is available: it reaches .798 pooled STATE/LAW balanced accuracy on the 303 recorded events, and LOCUSPROBE exceeds it by .067 [.017, .117]. On clean 8σ faults both variants pass a known-answer check. Per-testbed results for this variant were not computed, so it is not included in Table 2.

H RECORDED-VERSUS-INJECTED DETAILS

Under synthetic-only training the probe reaches .610 pooled balanced accuracy on the recorded events, against .516 for TabPFN v2, .512 for Toto 2.0 and .336 for MOMENT-1-large. Recorded labels are assigned by a numerical reference, with 90.8% agreement under an independent reference family.

On the residual-preserving grid an exact sign test gives $p = 0.956$, and 9 of 44 cells are significant under the less conservative Benjamini–Hochberg correction. Per testbed the mean gap is -0.149 (SKAB), -0.064 (HAI 21.03), $+0.005$ (ALFA), and $+0.071$ (Exathlon).

Table 10: Selected recorded-fault comparisons. Positive gaps favor injected anomalies. Intervals are reported class-standardized clustered-bootstrap 95% intervals where shown. Selected cells are illustrative; the full campaign count is $32/44$ positive and $18/44$ significant. The last two rows are the pooled template-clean estimate and the mean residual-preserving gap over all 44 cells.

Source	Model or summary	Injection-minus-recorded gap
SKAB extended	Qwen2.5-32B	+.418
Exathlon	Qwen3-14B	+.330 [.23,.44]
HAI 21.03	Qwen3-14B	+.224 [.08,.38]
ALFA	Across models	Positive in 8/11 cells
All four	Pooled, 11 models	+.135 [.094,.176]
All four	Residual-preserving twins	−.034 [−.078,+.008]

Matching controls. The matching construction holds normalized evidence magnitude $|\Delta|/\sigma$ equal across arms. The Bayesian reference finds the injected twins harder, and the model gap per-

sists after restricting to reference-confident cases. In the reported 72B analysis it is positive under every cause type. The gap is therefore not driven by anomaly magnitude or by the reference model's difficulty ranking.

This suggests a concrete evaluation practice: assess cause attribution on matched recorded and injected cases, retain class-standardized and raw views, and test a consequence of the diagnosis. The downstream experiment supplies the final test in an action-oriented harness.

Recorded-fault labels. Each recorded event is aggregated so that reading k starts at the logged changepoint, and kept only if the change is visible in that channel. Its class is operational rather than documentary: the argmax of the random-walk reference posterior (Appendix B) on the real stream, retained only when that posterior is confident and not OBS. The absence of recorded OBS events is therefore a property of this filter, not of the testbeds. The label is robust to the choice of reference: an independent local-linear-trend reference family, stored on every item, reproduces it on 0.908 of events (SKAB 0.826, ALFA 0.831, Exathlon 0.953, HAI 0.953). Because the random-walk reference defines the labels, no arm built on that same reference is compared against it on these items. Documented fault mechanisms are used only to group events for the per-mechanism analysis.

Reference observer on the twins. The same reference also gates the confidence of injected twins. It reproduces the planted cause on only 0.750 of residual-preserving twins, because a twin keeps the recorded residuals while swapping the cause template, and those residuals still carry traces of the original cause. 90.2% of residual-preserving twins pass the 0.9 confidence gate. The residual-preserving grid in the main text applies no gate to either arm and still yields a mean gap of -0.034, so the gate cannot be the source of the template-clean advantage.

Models. Eleven checkpoints span 1.7B–72B and three families in the recorded-versus-injected grid: Qwen3-1.7B, Qwen2.5-3B, Qwen2.5-7B, Llama-3.1-8B, Qwen3-8B, Phi-4 (14B), Qwen3-14B, Qwen2.5-32B, Qwen3-32B, Llama-3.1-70B, and Qwen2.5-72B.

Class standardization. Class standardization remains consequential: raw gaps change sign relative to standardized gaps in 12 of 44 cells. We therefore report every comparison under a common class weighting, which fixes the target population. A benchmark intended to predict real diagnosis performance must specify both how anomalies are constructed and how their cause frequencies enter the score.

The distinction can be written explicitly. For class-specific accuracies a_c^{inj} and a_c^{rec} and common comparison weights w_c, a standardized gap has the form

$$\Delta_{\mathrm{std}} = \sum_c w_c \big(a_c^{\mathrm{inj}} - a_c^{\mathrm{rec}}\big), \qquad w_c \geq 0, \quad \sum_c w_c = 1. \tag{5}$$

The sum runs over classes present in both arms; because recorded events contain no OBS faults, OBS receives zero weight. Common weights separate performance differences from differences in class composition, and the clustered analysis preserves the matched construction while estimating uncertainty for the chosen common-weight comparison.

I ADDITIONAL ADAPTATION AND TASK RESULTS

We show that correct diagnosis improves action outcomes in the evaluated tasks: actions based on the true cause achieve an average score of 0.782, compared with 0.368 for actions based on an incorrect cause. Synthetic-only fine-tuning also improves end-to-end performance on the weather and industrial tasks. These results connect cause identification to practical consequences within an experimental task environment, extending the evaluation from recognizing what went wrong to acting effectively on that diagnosis.

The two sources compose. A language model can also use evidence a numerical observer cannot read: one maintenance-log sentence naming what was done at the onset reading, with no numeric content, moves the untuned base model from 0.357 to 0.693, while the observer is unchanged by construction. Giving the base model the observer's three probabilities as one extra sentence raises it

to 0.552 with LAW recall 0.412. Combining the observer's and the adapter's posteriors as a geometric mean improves on the observer alone in every measured quantity: worst case 13.51 against 14.98 (paired difference $[-2.68, -0.45]$), mean loss 5.57 against 5.68, mean regret 2.864 against 2.968, harmful-revision rate 0.235 against 0.250, and accuracy when committing 0.583 against 0.550.

Hard cause-label supervision also helps, reaching 0.811 overall and 0.648 on navigation in that comparison. The useful ingredient is evidence-related supervision; the evidence does not establish that soft posterior targets are uniquely necessary.

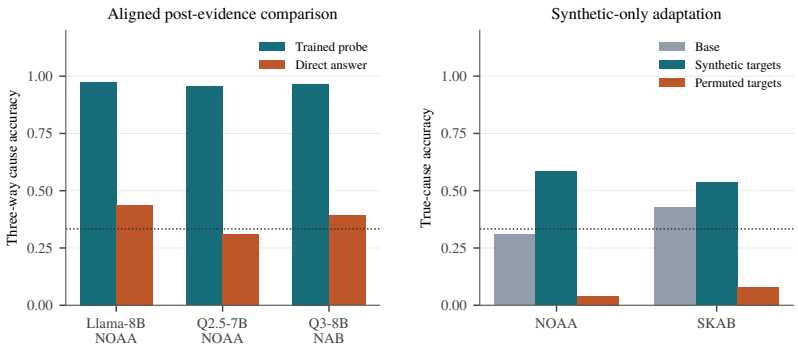


Figure 5: Two real-series comparisons. Left: three-way probe accuracy and direct-answer accuracy on aligned items; two direct-answer scores are approximate. Right: synthetic-trained adaptation and a matched fixed label-permutation control, scored against true causes. The dotted line is the uniform-choice reference $1/3$.

Table 11 gives the full synthetic adaptation comparison across four models. The transfer question is concrete: can training on arithmetic and logic improve identification of a changed movement rule? Qwen3-4B's accuracy on that navigation cause rises from 0.065 to 0.635 after posterior training.

At the ambiguous stage, the reported adapted entropy is 1.093, close to the uniform three-way value $\log 3 \approx 1.099$. Accuracy remains at the identical-input reference. The model becomes more decisive after resolving evidence without learning to select a confident explanation where the prompt supplies no distinguishing information. The informative calibration test is the intermediate stage, where the reference is neither uniform nor a point mass and the matched design constrains nothing. There, Qwen3-4B's probability agreement rises from 0.416 to 0.721 and its entropy falls from 1.075 to 0.565, toward the reference value of 0.157.

A separate synthetic next-reading question tests a consequence beyond the trained cause label. Qwen3-4B improves from 0.333 to 0.508 prediction accuracy and Qwen3-1.7B from 0.442 to 0.528. These improvements complement the real-series downstream experiments. The adapted forecasts also exceed what the base models achieve when given the cause: with the true cause stated in the prompt, base-model next-reading accuracy is only 0.367–0.482, and Qwen3-4B's is 0.367. On Qwen3-1.7B, MMLU and GSM8K slightly improve (0.584/0.794 adapted versus 0.573/0.778 base, on 800 and 500 questions), and on a separate counterfactual suite the adapter reduces anchoring to a stale derivation from 0.255 to 0.108.

The matched control keeps the adaptation recipe while rotating target labels by a fixed derangement. Scored against the true labels, it reaches only 0.040 on NOAA-based data and 0.080 on SKAB-based data, below uniform-choice accuracy. When scored against the rotated labels it was taught, it recovers to the semantic adapter's level. Thus the control learns the altered target relationship rather than simply failing to train.

Within the experimental arms, matching the true cause outperforms not matching it by a median 0.670 across 75 discriminating decisions. Parse failures are zero in all nineteen end-to-end cells.

The weather and industrial results supply a positive connection from learned diagnosis to task performance. A complete revision evaluation should ask whether the model identifies the cause, selects the appropriate update, and executes the resulting action. Each step can fail even when an earlier measurement improves.

Table 11: Original synthetic adaptation results. Navigation is held out from training. Qwen2.5-7B uses 540 evaluation instances; the other rows use 1,800. These cohorts are separate from the real-series adapter evaluation.

	All synthetic domains		Held-out navigation	
Model	Base	Adapted	Base	Adapted
Qwen3-1.7B	.349	.683	.333	.522
Qwen3-4B	.327	.782	.333	.595
Qwen2.5-7B	.459	.780	.397	.583
Llama-3.1-8B	.476	.783	.443	.632

Table 12: Downstream comparisons reported by the task harness. Each row is a separate comparison.

Comparison or error type	Task score or difference
Act under the true cause	.782
Act under a wrong cause	.368
Miss a trend or regime change	.017–.181
Confuse observation glitch and state shock	.20–.33
Weather end-to-end adapter advantage over base	+.147
Weather end-to-end advantage over permuted	+.105
Weather plug-in advantage over permuted	+.198
Weather uniform-target adapter relative to base	−.022
Industrial end-to-end adapter advantage over base	+.106
Industrial plug-in advantage over permuted	+.111

These experiments give cause attribution operational meaning. An observation error may disappear after a single step, whereas a missed rule change can continue to distort every later prediction. The 60-step rollouts make this difference explicit, while the separate action task shows that choosing the correct revision also improves decision quality. More generally, a diagnosis should be judged together with the update it triggers and the trajectory that follows.

Action-oriented task. A separate action-oriented task gives the same qualitative conclusion. Acting under the true cause gives mean score 0.782, versus 0.368 under wrong causes. The error type matters. Missing a genuine trend or regime break gives scores 0.017–0.181, compared with 0.20–0.33 for confusing an observation glitch with a state shock. A rule error can continue to distort future predictions rather than only mislocating one state.

The semantic adapter beats the identically trained permuted control by 0.198 on its task format, in the plug-in arm where the forecast is computed from the supplied diagnosis alone. In the weather end-to-end comparison, its reported advantage is 0.147 over base, with a label-specific advantage of 0.105 over the permuted control. The corresponding industrial comparison improves by 0.106 over base end to end; its 0.111 advantage over the permuted control is measured in the plug-in arm, so the industrial end-to-end gain is not attributed to label content. Table 12 summarizes these comparisons. The uniform-target adapter scores 0.363 on the weather task, below the base score of 0.385, so extra fine-tuning alone does not produce the gain. Attribution gains and action gains must therefore remain separate endpoints.

J Interpreting adaptation targets

Matched hard-label tuning reaches .830 intermediate agreement and .856 resolved accuracy, and readout-only training reaches .863 LAW recall while leaving intermediate agreement at .428.

The exact-posterior objective provides targets for both partial and resolved evidence. It specifies the desired uncertainty at every stage, and with navigation held out it produces the reported gains. Hard labels are also effective, as described next.

Hard-label training also preserves ambiguity in the controlled design. Identical initial prompts occur with balanced causes, so the data themselves discourage a consistently confident choice at that stage.

For Qwen3-4B, ambiguous entropy is 1.091 under hard labels. Exact posterior targets express the uncertainty explicitly; hard labels can convey it through the balanced construction. This explains why the semantic-target comparison supports more than one useful supervision rule.

Generic benefits of extra fine-tuning cannot explain the true-label gains: format familiarity or a shift in letter preference cannot make performance follow the supplied semantic mapping. The gains come from evidence-to-cause targets.

The synthetic supervision ablation permutes the posterior by a cyclic shift of one or two positions chosen per triplet, which cannot be recovered from the prompt; a learner that matches it aggregates to chance against true labels, as observed (0.333). The real-series control instead uses one fixed cyclic relabeling for all instances, so a learner that matches it is consistently anti-correlated with the truth, as observed in the main text.

K Reported priors and per-cause costs

The apparent uncertainty at the first stage is a prior. All 600 triplets give byte-identical prompts at that stage, so the answer there cannot depend on the evidence, and the models largely keep reporting it: an agent that repeats each model's first-stage answer at every later stage reproduces Qwen3-4B's probability agreement to within 0.014. The prior differs by model family. Whatever actually happened, Qwen3-4B names an observation fault in about nine of ten resolved examples. Tests of independence between the true cause and the answer cannot reject independence for two of five configurations, and where they reject it the association is negligible (Cramér's $V \leq 0.095$), except for Qwen2.5-7B ($V = 0.272$). The Qwen3 family labels 0.903 and 0.663 of changed-rule examples as observation faults, whereas Qwen2.5-7B labels 0.573 of observation faults as rule changes (Appendix L). Missing a changed rule is the error that Section 3 shows compounds across future predictions.

Restricted to the 120 items on which Qwen2.5-72B's direct answers exist, and refitted under the same layer-16, triplet-grouped folds, the Llama-3.1-8B probe reaches 0.967 against the 72B model's 0.842, despite using a model nine times smaller; comparing a 600-item probe with a 120-item answer would not be a comparison. The contrast is especially striking because Llama-3.1-8B's own answers achieve only 0.438 accuracy, revealing how much diagnostic information a supervised classifier can recover beyond what the model reports directly. Larger models can also use new evidence effectively, as Qwen2.5-72B improves from 0.33 to 0.84 as observations arrive. These results suggest that improving how existing representations are used can offer substantial gains alongside increasing model size, while preserving the distinction between a trained classifier's predictions and the language model's own answers.

The per-cause results explain this difference. For the Qwen3-8B adapter, an OBS fault gives errors of 7.80, 8.62, and 7.80 under oracle revision, model-selected revision, and no revision, respectively. A single incorrect observation therefore largely disappears without intervention. For a STATE disturbance, the corresponding errors are 6.83, 37.03, and 8.07, showing that an incorrect revision can be worse than leaving the system untouched. The largest asymmetry occurs for LAW changes: oracle and model-selected revision both reach 19.84, while no revision reaches 302.46. An uncorrected rule continues to affect each subsequent transition, so its prediction error compounds across the rollout.

L Synthetic benchmark details

Across the six configurations the best model, Qwen2.5-7B, reaches 0.459 at the resolved stage. The floor is not an arithmetic limit: on logic, which requires no numeric simulation, the sub-7B models also remain at 0.32–0.38. Scale does not remove it either: Qwen3-8B stays at 0.335, Qwen3-14B and Qwen3-32B reach 0.434 and 0.431 (4-bit), and Llama-3.1-8B and Phi-4 reach 0.476 and 0.437 on the full set.

Evidence sensitivity. Median belief displacement between the ambiguous and final stages is 0.036–0.095, with argmax changes in 3–9% of examples. The exception is Qwen2.5-7B, whose displacement is larger (0.230) and depends on the true cause (0.378 for LAW versus 0.223 for STATE).

Table 13: Answers on compound and off-taxonomy inputs, as proportions over OBS/STATE/LAW ($n = 120$ per cell). Each model answers with its own constant on every condition, including inputs with no fault.

Condition	Qwen3-4B	Qwen2.5-7B
OBS + STATE	.00 / .00 / **1.00**	.07 / **.93** / .00
STATE + LAW	.00 / .00 / **1.00**	.12 / **.88** / .00
slow drift	.00 / .00 / **1.00**	.03 / **.97** / .00
no fault at all	.00 / .00 / **1.00**	.01 / **.99** / .00

Table 14: Synthetic benchmark as evidence arrives (j = disambiguating observations seen; 1,800 examples). Agreement is the probability agreement of Section 2.4; H is belief entropy. The two Qwen3-1.7B modes are the same computation under the immediate-answer readout and are shown once. Bootstrap 95% half-widths over triplets are ≤ 0.03 for every accuracy. Under deterministic execution the j=0 accuracy is exactly $1/3$.

	Cause accuracy			Agreement		Entropy H	
Model	j=0	1	2	j=0	2	j=0	2
Qwen3-4B	.332	.336	.327	.91	.35	1.07	1.08
Qwen3-1.7B	.333	.344	.349	.81	.34	1.00	0.99
R1-Distill-1.5B	.334	.334	.336	.89	.34	1.07	1.07
Qwen2.5-3B	.333	.345	.344	.82	.34	0.99	1.01
Qwen2.5-7B	.332	.380	.459	.72	.39	0.87	0.94
Reference	.333	.889	1.000	1.00	1.00	1.10	0.00

An always-OBS agent scores 0.333 at every stage. The difference in probability agreement between the model and a replay of its own first-stage answer is at most $+0.014$ for Qwen3-4B, and -0.001 $[-0.001, +0.000]$ for Qwen3-1.7B at j=1 with a triplet-level bootstrap interval; Qwen2.5-7B deviates from its replay by $+0.052$.

Determinism and prior sensitivity. Under batch size one, TF32 disabled, and deterministic kernels, the largest within-triplet total-variation distance on byte-identical prompts is 0.00000 over 201 triplets for both Qwen2.5-7B and Qwen3-4B. Halving or doubling the prior mass of any cause class changes model probability agreement by at most 0.013 and leaves reference accuracy unchanged to three decimals. A quantization check on real data bounds 4-bit versus bf16 differences at ≤ 0.05 with ordering unchanged; the 4-bit Qwen3-14B and Qwen3-32B values are measured on a stratified 600-example subsample.

M DELIBERATION AND PROMPTING CONTROLS

On real NOAA items no elicitation closes the gap: chain-of-thought raises Qwen2.5-7B from 0.307 to 0.517, the largest gain observed, and thinking mode reaches 0.489.

A time-series LLM does not report rule changes. ChatTS-14B (Xie et al., 2025), a language model with a native time-series encoder, receives the same items, byte-identical prompts, and the same readout; we report its stronger text-only arm. On the 300-item NOAA attribution set it reaches 0.417 with LAW recall 0.080, against 0.307 for the base Qwen2.5-7B and 0.597 for its synthetic-trained adapter. Through its native encoder it commits to a rule change in none of the 400 revision episodes and scores 0.341 on recorded SKAB faults.

Allowing the model to reason does not close the gap. At the final stage, explicit reasoning changes accuracy by at most three points, and Qwen3-32B with thinking enabled reaches only 0.269 at the final stage, at or below the $1/3$ floor. Reasoning traces usually simulate a hypothesis and state a decision rule; what fails is the assignment of the error to a cause (Appendix M).

The adapter also commits sooner than the numerical observer on the revision benchmark, reaching confidence 0.6 after 1.14 evidence steps against 1.24, and failing to commit on 0.01 of episodes

Table 15: Mean readout at the ambiguous stage, where the three members of a triplet are the same prompt, so the readout is a prior rather than an inference. PCI $= 1 - \mathbb{E}_i\big[H(P_M^{(i)})\big] / \log 3$ is the mean over examples i of per-example premature commitment. Qwen2.5-7B's mean belief is near-uniform while its per-example beliefs are not, which is why its PCI exceeds the others'.

Model	$P_M(\text{OBS})$	$P_M(\text{STATE})$	$P_M(\text{LAW})$	PCI
Qwen3-4B	.422	.303	.276	.022
Qwen3-1.7B	.491	.321	.188	.093
R1-Distill-1.5B	.382	.314	.304	.029
Qwen2.5-3B	.226	.444	.330	.094
Qwen2.5-7B	.344	.315	.341	.210
Reference	.333	.333	.333	.000

Table 16: Error types at the final stage, where the reference is a point mass. Missing a rule change is LAW→OBS; treating an observation fault as a rule change is OBS→LAW; treating a local state disturbance as a rule change is STATE→LAW. The test is of independence between the true cause and the answer.

Model	Accuracy	LAW→OBS	OBS→LAW	STATE→LAW	χ^2 p	Cramér's V
Qwen3-4B	.327	.903	.027	.010	1.4×10^{-6}	.095
Qwen3-1.7B	.349	.663	.000	.002	.214	.040
R1-Distill-1.5B	.336	.337	.332	.330	.995	.008
Qwen2.5-3B	.344	.065	.273	.272	.007	.063
Qwen2.5-7B	.459	.217	.573	.517	$< 10^{-10}$	.272

against the base model's 0.50. Under *ignore*, the measured rule-change to state-shift cost ratio grows 1.04, 1.70, 2.96, and 5.14 at horizons 5, 10, 20, and 40.

Table 17: Explicit reasoning before answering versus the immediate-answer readout on a 126-example subsample. The immediate-answer columns are recomputed on the same subsample. Generations that commit to an answer are scored (parse rates 0.93–1.00).

	Explicit reasoning			Immediate answer		
Model	j=0	1	2	j=0	1	2
Qwen3-1.7B	.336	.344	.376	.333	.344	.349
R1-Distill-1.5B	.378	.385	.356	.334	.334	.336
Qwen2.5-7B	.254	.437	.452	.332	.394	.442

Trace coding. We code 200 reasoning traces at the final stage with a lexical coder and a Qwen2.5-14B judge: 61–87% of traces forward-simulate at least one hypothesis, 68–96% cite the post-anomaly observations, 81–91% state a decision rule, and only 0.5% merely restate the prompt and guess. On arithmetic, where each hypothesis implies a numeric continuation not present in the prompt, only 23–55% of traces contain even one correctly computed continuation and 0–7% contain all of them.

Prompting conditions. Five worked examples change final-stage accuracy by at most $+0.09$, and by -0.01 for Qwen3-1.7B. Appending one generated sentence that states what the post-anomaly readings show, leaving only the step from evidence to cause, also helps by at most $+0.09$. A scaffold that instructs the model to predict the continuation under each cause, compare, and then commit is the only intervention that moves a model substantially: Qwen2.5-7B reaches 0.665 overall and 1.00 on arithmetic (on the 51% of generations that commit), while Qwen3-1.7B improves by only 0.10. The adapter instead improves attribution in a single forward pass without generated simulation.

N HIDDEN-STATE CLASSIFIER CONTROLS

Capacity control. The identical classifier trained on random per-triplet labels, with the same layer, folds, and regularization, is at chance at every layer (0.29–0.33), while the task classifier is well above chance under the same pipeline. The pipeline therefore does not produce above-chance accuracy from arbitrary labels. Logistic regression on the raw standardized readings reaches 0.518 and on hand-crafted summary features 0.493, both near the surface-token level.

Synthetic cross-domain transfer. A classifier trained on arithmetic only transfers to navigation at 0.468 and to logic at exactly 0.333. Transfer therefore follows shared numeric structure, consistent with the synthetic-to-real transfer in the main text. Leave-one-domain-out transfer for Qwen3-4B is 0.375, 0.333, and 0.510 for held-out arithmetic, logic, and navigation.

Activation steering. We add the unit class direction of the classifier, scaled as a percentage of the mean residual norm, at the readout position of the probed layer and measure the answer distribution on 300 final-stage examples (Qwen3-4B). Probability on the steered cause rises monotonically with dose (0.340, 0.349, 0.362, 0.372 at 4, 8, 16, and 80%), symmetrically whether the steered cause is true or false. Random directions of matched norm shift probability by only +0.013–0.015. The direction therefore carries cause-specific content that moves the model's answer.

The diagnostic direction is nearly orthogonal to the answer directions. The probe reads cause information along a direction in hidden-state space that is only weakly aligned with the directions used to score the answer letters. Their maximum absolute cosine similarity is 0.02–0.04, close to the random-direction reference of approximately 0.02. Diagnostic information is thus accessible to a trained classifier without being reflected in the model's immediate answer scores.

Activation patching. In Qwen2.5-7B ($n = 60$ triplets), we inject the donor residual stream at the last prompt token of the matched OBS member and measure the change in $P(\text{LAW})$, whose baseline on these items is 0.654. Table 18 reports three donor conditions. The cause patch is large only at later layers, the same-cause donor stays within ± 0.03, and a donor that has not yet seen the resolving evidence pushes the output the other way. The cause signal is therefore carried by the later layers and depends on the resolving evidence.

Table 18: Change in $P(\text{LAW})$ on OBS items after residual patching at the last prompt token (Qwen2.5-7B, $n = 60$ triplets).

Layer	LAW→OBS cause patch	Same-cause donor	Pre-resolution donor
0–16	+.001 to +.012	+.002 to +.011	+.002 to +.008
20	+.116	−.030	−.095
24	+.119	+.010	−.099
27	+.120	+.000	−.106