EVIDLOOP: STAGED VERIFICATION FOR AGENTIC MODEL SEARCH IN AERODROME WEATHER FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-model agents increasingly turn research hypotheses into executable model edits, but improvements observed during adaptive search may not persist across training seeds or evaluation conditions. We introduce EvidLoop, a framework that links each evaluated implementation to staged acceptance criteria and separates development feedback from final assessment. Candidates advance through single-seed screening, paired five-seed confirmation, and one-time assessment of frozen models on temporal and spatial holdouts, including a predefined weather-event subset termed HAZARD. On the AeroWF aerodrome-weather benchmark, three language models explore four intervention streams, yielding 70 scored attempts, 24 screening survivors, and 11 confirmed candidates. The confirmed edits reduce development RMSE by 1.22%–3.76% relative to the parent's mean development RMSE. Seven satisfy a joint mean-based acceptance rule, fixed before held-out evaluation, that requires temporal improvement and limits spatial and HAZARD-conditioned degradation. Five improve all three held-out means. Four of the eleven confirmed candidates improve temporal and spatial means but exceed the HAZARD margin, including the candidate with the largest temporal gain. Accepted edits span error-dependent reweighting, horizon-conditioned robust penalties, and adaptive residual scaling. These results identify useful forecasting modifications and show that cross-seed development gains and aggregate held-out improvements can coexist with conditional regressions. EvidLoop makes the resulting acceptance decisions explicit and traceable to the implementations evaluated.

1 INTRODUCTION

Language-model agents increasingly turn research ideas into executable experiments. The AI Scientist, Agent Laboratory, and Dolphin connect hypothesis generation, implementation, and experimental feedback in automated or assisted research workflows (Lu et al., 2024; Schmidgall et al., 2025; Yuan et al., 2025), while EvoPrompting extends this capability to code-level model design (Chen et al., 2023). Closed-loop Auto Research for Molecular Property Prediction further connects adaptive interventions to held-out assessment (Ning et al., 2026a). Together, these advances make agentic model search a practical setting for studying how proposed improvements become verified results.

Execution studies show that favorable assessments of LLM-generated research ideas need not translate into equally favorable experimental outcomes (Si et al., 2026). For adaptive model search, this raises a candidate-level question: when should an executable edit be accepted as an improvement? A proposal may undergo several code revisions, so experimental evidence must be linked to the implementation actually evaluated. A favorable screening result may not persist across training seeds, while aggregate held-out gains may coexist with degradation under specific conditions. Search also depends on which outcomes the agent can observe: development feedback guides exploration, whereas final assessment must retain a separate role. An acceptance procedure must therefore specify the candidate to which evidence belongs, the decision supported by that evidence, and whether it may guide search.

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.

We introduce *EvidLoop*, a closed-loop search framework that jointly specifies these three elements. Executable identities bind experimental evidence to individual edits; stage-specific criteria govern screening, paired cross-seed confirmation, and frozen held-out acceptance; and feedback boundaries separate adaptive development from final assessment. Agents explore permitted modifications under a fixed parent specification and evaluation contract. The resulting candidate records preserve both useful discoveries and the evidence explaining why other edits fail to advance.

We instantiate EvidLoop on AeroWF, a multi-airport aerodrome-weather forecasting benchmark (Li et al., 2026). Three language models explore representation, downstream-objective, pretraining-objective, and model interventions. Final assessment covers future periods at development airports, an unseen airport, and a HAZARD-conditioned slice of both holdouts. The joint acceptance rule, fixed during discovery and before sealed evaluation, requires temporal improvement, bounded spatial and HAZARD-conditioned mean degradation, and valid primary predictions.

The frozen campaign yields 70 scored attempts, 24 screening survivors, and 11 confirmed candidates. All 11 improve temporal and spatial means, but four exceed the HAZARD margin, including the candidate with the largest temporal improvement. Seven satisfy the full acceptance rule, and five improve all three held-out means. The accepted edits span error-dependent reweighting, horizon-conditioned robust penalties, and adaptive residual scaling. Thus, the campaign both discovers useful objective modifications and demonstrates why cross-seed development gains and aggregate held-out improvements are insufficient to determine conditional acceptance.

Our contributions are threefold:

- **A candidate-level acceptance framework for agentic search.** EvidLoop combines executable identity, stage-specific evidence requirements, and feedback boundaries to govern how an edit advances from adaptive discovery to final acceptance.
- **An empirical account of what successive verification stages establish.** A frozen campaign across four intervention streams distinguishes screening gains from valid, consistent cross-seed improvement, and aggregate held-out gains from conditional acceptance, while retaining refuted, invalid, and unresolved records.
- **Accepted objective modifications for aerodrome forecasting.** Seven candidate records satisfy joint held-out acceptance. We characterize their reweighting, robust-penalty, and residual-scaling implementations through measured horizon profiles and held-out effects.

2 Related Work

Research agents and executable search. The AI Scientist, Agent Laboratory, and Dolphin connect hypothesis generation, implementation, and experimental feedback in research workflows (Lu et al., 2024; Schmidgall et al., 2025; Yuan et al., 2025); the AI Scientist-v2 further organizes experimentation through agentic tree search (Yamada et al., 2025). FunSearch and AlphaEvolve couple code generation with evaluator-guided search (Romera-Paredes et al., 2024; Novikov et al., 2025), while EvoPrompting and AgentHPO target architecture and hyperparameter optimization (Chen et al., 2023; Liu et al., 2025). MLE-bench, ScienceAgentBench, and RE-Bench assess executable outcomes in engineering and scientific tasks (Chan et al., 2025; Chen et al., 2025; Wijk et al., 2025). EvidLoop focuses on the evidence required to accept individual edits from such processes.

Closed-loop discovery and held-out verification. Closed-loop Auto Research for Molecular Property Prediction and Auto Research for Materials separate adaptive discovery from held-out evaluation of executable interventions (Ning et al., 2026a;b). AgentFold combines experimental feedback with structured intervention memory to search protein-folding model variants (Liu et al., 2026). Building on this separation between search and final assessment, EvidLoop focuses on candidate-level acceptance: evidence must remain attributable to the evaluated implementation, development gains must satisfy paired cross-seed requirements, and held-out performance must meet both aggregate and event-conditioned criteria. Our campaign examines the disagreements between these stages, including edits that improve across training seeds and both aggregate holdouts yet fail the event-conditioned requirement. This focus complements established work on model-selection bias and adaptive evaluation (Cawley & Talbot, 2010; Dwork et al., 2015; Nakkiran & Błasiok, 2018).

Objective design and aerodrome forecasting. Deep learning has been investigated for aviation wind forecasting (Alves et al., 2024), and AeroWF provides the multi-airport setting used here (Li et al., 2026). Our discovered objective edits draw on established robust-loss ideas (Barron, 2019). Their evaluation also connects to work on real-world distribution shifts (Koh et al., 2021) and hidden stratification, where aggregate performance can obscure important subgroup failures (Oakden-Rayner et al., 2020). EvidLoop assesses both aggregate transfer and a predefined HAZARD-conditioned slice, connecting objective modifications to their measured behavior.

3 EvidLoop: Closed-Loop Search with Staged Verification

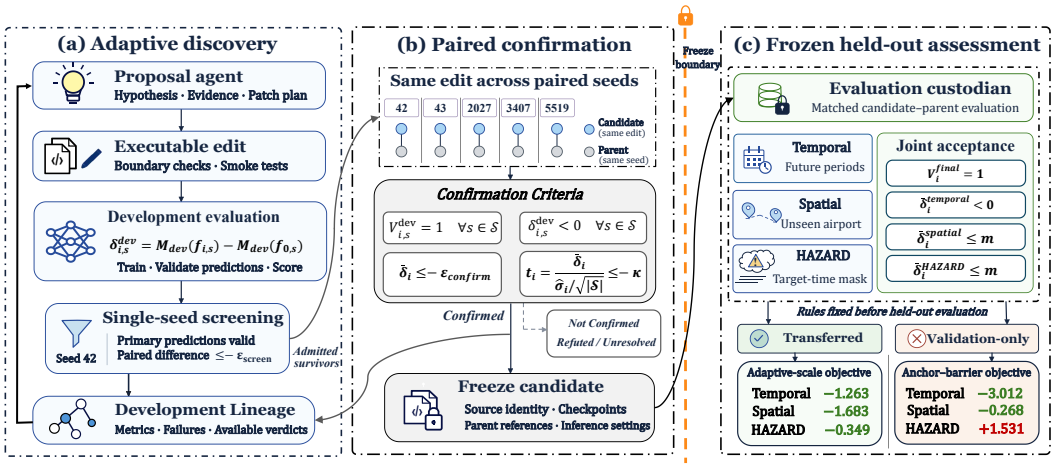


Figure 1: EvidLoop: closed-loop discovery with staged verification. (a) Agents propose executable edits using available development feedback; single-seed screening selects candidates for confirmation. (b) The same edit is evaluated across five paired seeds, and confirmed candidates are frozen. (c) Joint held-out acceptance assesses temporal and spatial performance together with their HAZARD-conditioned slice. Held-out results never feed back into search. The examples show five-seed mean paired normalized-RMSE differences ($\times 10^{-3}$) for the adaptive-scale edit (ID 976) and robust anchor–barrier edit (candidate #982): stronger temporal improvement need not imply joint acceptance.

3.1 Framework Overview

EvidLoop evaluates executable edits through screening, paired cross-seed confirmation, and frozen held-out assessment (Figure 1). Each search campaign specifies a parent model, training procedure, permitted edits, execution budget, and evaluator. A candidate record links the implemented edit to its experimental results and determines which evidence can support its advancement.

Agents use development history—previous proposals, executed edits, and available experimental results—to guide subsequent proposals. Confirmation evaluates the same implementation across training seeds; a code revision requires new evaluation evidence. After discovery closes, confirmed candidates are frozen and assessed by an evaluation custodian who controls access to the held-out data. The final results determine acceptance and are not returned to the agents as search feedback.

3.2 Evidence-Guided Proposals and Candidate Identity

At proposal time, the agent receives the campaign specification and development evidence available within its lineage. The context combines persistent campaign facts, records relevant to the active intervention stream, and summaries of other streams. Available observations can include metrics, prediction-validity outcomes, execution failures, and completed confirmation results. A screening result may therefore inform a proposal before its confirmation verdict becomes available.

Proposal generation and code generation are separate steps. A structured proposal specifies a mechanism hypothesis, supporting evidence, an expected effect, a falsification condition, and a patch plan. The agent then implements the edit within the permitted files and interfaces. The harness checks these boundaries, performs smoke tests, executes the prescribed training pipeline, and restores editable files afterward. The agent cannot edit the evaluator, private targets, or rules.

Experimental evidence comprises execution outcomes, prediction-validity observations, and paired performance differences when available. A candidate record links these observations to the executed edit, proposal, parent reference, evaluation configuration, and seeds. Source, checkpoint, and prediction identifiers establish provenance; performance observations support decisions about improvement. Confirmation uses the same executed edit as screening; a code change requires new evaluation evidence. Development observations can inform later proposals only after they become available, whereas held-out observations serve final assessment. The prescribed parent reference remains fixed.

The evaluator checks coverage, uniqueness, shapes, finite values, allowed ranges, and reproduction anchors before scoring predictions under fixed aggregation rules. It masks unavailable targets and padded positions without silently repairing predictions. Validity is resolved separately for each task output: an invalid auxiliary output need not invalidate the primary forecasting result, whereas invalid primary predictions cannot support advancement to the next verification stage.

3.3 Screening and Paired Cross-Seed Confirmation

For candidate i, training seed s, and evaluation condition e, let $V_{i,s}^{e}$ indicate the validity of the required paired forecasting predictions. When valid, the candidate's effect is

$$\delta_{i,s}^{e} = M_e(f_{i,s}) - M_e(f_{0,s}), \tag{1}$$

where M_e is the evaluation metric and $f_{0,s}$ is the same-seed parent reference prescribed by the comparison contract. Negative differences favor the candidate.

Screening. A candidate qualifies for additional evaluation when its primary predictions are valid and its paired difference meets the screening threshold:

$$V_{i,s_0}^{\mathrm{dev}} = 1, \qquad \delta_{i,s_0}^{\mathrm{dev}} \le -\epsilon_{\mathrm{screen}}. \tag{2}$$

Screening eligibility and completion of confirmation are recorded separately.

Paired confirmation. Screening survivors selected for confirmation are evaluated with the same executable edit on the additional prescribed seeds. Let $\mathcal{S}$ be the full seed set, including s_0, and let $\bar{\delta}_i$ and $\widehat{\sigma}_i$ denote the mean and sample standard deviation of the candidate's paired development differences. Confirmation requires

$$\begin{aligned} &V_{i,s}^{\mathrm{dev}} = 1 \quad \text{and} \quad \delta_{i,s}^{\mathrm{dev}} < 0 && \forall s \in \mathcal{S}, \\ &\bar{\delta}_i \le -\epsilon_{\mathrm{confirm}}, \qquad t_i = \frac{\bar{\delta}_i}{\widehat{\sigma}_i / \sqrt{|\mathcal{S}|}} \le -\kappa. \end{aligned} \tag{3}$$

These conditions check validity, directional consistency, mean improvement, and seed variation.

A required validity or sign failure permits early termination once confirmation becomes unattainable. A candidate failing the rule receives a *refuted* verdict; one without a completed decision at the cutoff remains *unresolved*. Partial observations and stopping reasons are retained.

Confirmation remains part of adaptive development: it includes the screening seed, and completed outcomes may inform subsequent proposals. Its criteria govern candidate advancement rather than provide an independent post-selection significance test.

3.4 Frozen Held-Out Assessment and Joint Acceptance

After discovery closes, all confirmed candidates are frozen with their source identities, seed-specific checkpoints, parent references, and inference specifications. The custodian evaluates matched candidate and parent models on sealed data. Acceptance criteria must be fixed before sealed evaluation; the time at which they were fixed is recorded separately from the candidate-freeze time.

For AeroWF, final assessment comprises temporal and spatial holdouts and an event-conditioned slice of predictions from both. Let $\bar{\delta}_i^e$ be the mean paired difference across the prescribed seeds and V_i^{final} indicate that all required final validity checks pass. Joint acceptance requires

$$\begin{aligned} &V_i^{\text{final}} = 1, \qquad \bar{\delta}_i^{\text{temporal}} < 0, \\ &\bar{\delta}_i^{\text{spatial}} \le m_{\text{spatial}}, \qquad \bar{\delta}_i^{\text{event}} \le m_{\text{event}}. \end{aligned} \tag{4}$$

The margins bound permitted mean degradation. Improvement under one evaluation condition cannot compensate for failure under another.

Candidates satisfying the mean-based rule are accepted (recorded as *transferred*); completed assessments that fail it remain *validation-only*, and incomplete assessments remain *inconclusive*. Final evaluation results do not inform further search.

4 Experimental Setup

We evaluate EvidLoop on AeroWF (Li et al., 2026) through four research questions: **RQ1:** What forecasting improvements do the discovered edits achieve over the parent? **RQ2:** How does staged verification distinguish screening gains from confirmed and accepted candidates? **RQ3:** How do HAZARD-conditioned performance and the degradation margin affect held-out acceptance? **RQ4:** Which objective designs satisfy the verification criteria, and what do supporting controls reveal about search?

4.1 Task, Data, and Evaluation Metrics

Task and data. AeroWF contains runway observations from four airports. Each input comprises 96 one-minute observations, with consecutive window starts separated by 15 minutes. The forecasting task predicts two horizontal wind components at 15, 60, and 120 minutes after the final input observation. ZBAA, ZSPD, and ZSSS provide training, development (validation), and temporal holdout data; ZBAD is reserved for spatial evaluation. After boundary-window removal, these partitions contain 56,598, 11,883, 12,198, and 27,101 input windows, respectively. Normalization is fitted on training data, and only targets within the assigned split are scored. The forecasting loss uses complete-horizon anchors, yielding 56,574 training and 11,859 validation samples. Prediction export retains all anchors and masks unavailable targets and padded runway positions.

Metrics and evaluation conditions. The primary metric is normalized wind RMSE, computed within each airport–horizon–component endpoint and averaged equally across endpoints. Development and temporal evaluation each comprise 18 endpoints, while spatial evaluation comprises six. The HAZARD readout applies a frozen target-time event mask to both holdouts and aggregates 24 endpoints. It is an event-conditioned evaluation of the two holdouts, rather than an additional independent test set. Weather classification supplies auxiliary diagnostics. Appendix A details data construction, event support, and METAR timestamp provenance.

4.2 Parent Model and Training Configuration

Parent model. The parent combines temporal and frequency encoders, a hierarchical graph module, and a METAR encoder through residual fusion and a forecasting head. Its forecasting optimizer updates approximately 2.77 million parameters. We measure candidate effects as normalized-RMSE differences from the prescribed same-seed parent, with negative differences indicating improvement.

Training configuration. Downstream training uses AdamW for 60 epochs, batch size 128, cosine learning-rate decay to zero, and the final checkpoint without early stopping. Scratch runs use a learning rate of 10^{-4}. Pretrained runs use 3×10^{-5} for the encoder core and 10^{-4} for the head. Each training run uses one RTX 3090 in FP32. Full architecture, initialization, and optimization settings are provided in Appendix G.1.

Comparison contracts. Representation and downstream-objective candidates are screened against scratch-trained parents using scratch-trained candidate models. Pretraining-objective and some model interventions instead use pretrained candidates against scratch parents. Each record retains its prescribed comparison contract through evaluation and reporting.

4.3 Search Campaign and Reporting Units

DeepSeek, GLM, and Kimi maintain separate development histories under a common parent specification and evaluator. The main search campaign contains 70 scored attempts: 28 representation, 19 downstream-objective, seven pretraining-objective, and 16 model interventions. The candidate set was frozen on September 13, 2026.

The scored set contains 57 completed and 13 scored-invalid records, with ten additional execution failures outside the scored denominator. Eligibility follows primary forecasting validity: an auxiliary-output failure does not necessarily invalidate the forecasting result. Appendix C reports the complete accounting, exclusions, and implementation identifiers.

4.4 Verification Criteria and Statistical Reporting

A horizon-weighted MSE probe with weights $(3, 1, 1)$, evaluated on seeds 42, 43, 2027, 3407, and 5519, provides the paired-difference scale $m = 0.000287$. Screening uses seed 42 and threshold $-m$. Five-seed confirmation uses mean threshold $-2m = -0.000574$ and paired-statistic threshold -2.776, together with the validity and sign requirements in Section 3. The held-out margins in Eq. (4) are $m_{\text{spatial}} = m_{\text{event}} = m$.

Development criteria preceded the first recorded main-campaign result; the custodian signed off on final acceptance criteria during discovery, before candidate freezing and held-out evaluation. We report five-seed means and paired differences, with pointwise paired-seed intervals in Appendix D. Appendix B documents calibration and policy chronology.

4.5 Supporting Controls and Analyses

A shared finite-template reference contains 28 completed trials covering representation and downstream-objective edits under the same parent, evaluator, and per-trial training budget. Its coverage, repair opportunities, and validity accounting differ from the LLM campaign. The sole screening survivor, weighted MSE with weights $(2, 1, 1)$, received supplementary five-seed confirmation.

A development-history control compares four proposal rounds with and without history in each of four paired blocks, using the same Kimi model, parent, editing interface, and screening metric. Its 32 slots, detailed in Appendix F, comprise 27 valid results, two invalid executions, and three proposal timeouts before training.

Additional analyses examine horizon effects, confirmation without the screening seed, and HAZARD-margin sensitivity. Recorded discovery and control executions total approximately 862 RTX 3090 wall-clock GPU-hours; see Appendix G.6.

5 Results

5.1 Main Forecasting Results (RQ1)

EvidLoop discovers eleven confirmed edits, seven of which satisfy joint held-out acceptance (Table 1). All eleven compare candidate and parent models trained from scratch. Their mean development RMSE ranges from 0.047499 to 0.048753, versus the parent's 0.049356, a reduction of 1.22%–3.76%. All eleven improve temporal and spatial means, and five also reduce mean RMSE on the HAZARD slice.

The adaptive-scale pseudo-Huber edit (candidate #976) achieves the lowest development RMSE and largest spatial gain, whereas the horizon-conditioned pseudo-Huber edit (candidate #885) has the largest HAZARD improvement. The strongest temporal gain belongs to the anchor–barrier edit (candidate #982), which fails joint acceptance.

Table 1: Forecasting results of EvidLoop-discovered edits. All scores are five-seed means in units of 10^{-3} normalized RMSE. Development reports absolute scores; held-out columns report paired differences from the same-seed parent (negative is better). Accepted denotes satisfaction of the joint held-out rule. Bold marks the lowest reported mean in each column.

Objective edit (ID)	Dev. RMSE ↓	Temporal Δ	Spatial Δ	HAZARD Δ	Accepted
Parent	49.356	0	0	0	–
Development-only references					
Weighted MSE $(3, 1, 1)$	48.440	–	–	–	–
Weighted MSE $(2, 1, 1)$	48.870	–	–	–	–
EvidLoop: all confirmed candidates					
Horizon reweighting (741)	48.753	-0.440	-0.726	$+0.208$	Yes
Cell reweighting (763)	48.246	-0.446	-1.047	$+0.249$	Yes
Horizon-robust loss (873)	47.875	-1.024	-0.884	-0.507	Yes
Horizon-robust loss (883)	47.809	-1.251	-0.951	-0.808	Yes
Horizon pseudo-Huber (885)	47.797	-1.133	-1.101	$\mathbf{-0.880}$	Yes
Robust loss + range penalty (928)	47.682	-1.174	-1.224	-0.661	Yes
Adaptive cell weighting (939)	48.673	-0.073	-0.604	$+0.398$	No
Range-barrier edit (943)	48.323	-0.613	-0.825	$+0.957$	No
Range-barrier edit (945)	48.323	-0.613	-0.825	$+0.957$	No
Adaptive-scale pseudo-Huber (976)	**47.499**	-1.263	$\mathbf{-1.683}$	-0.349	Yes
Anchor–barrier edit (982)	48.252	$\mathbf{-3.012}$	-0.268	$+1.531$	No

IDs link candidates to the figures and archived implementations. The fixed-weight references are the calibration probe $(3, 1, 1)$ and the random-template survivor $(2, 1, 1)$. Dashes denote unavailable held-out results or inapplicable status; parent zeros denote reference differences.

Table 2: Evidence supporting decisions at successive stages. Differences are normalized RMSE in units of 10^{-3}; held-out differences are five-seed means. Confirmation requires a mean of at most -0.574; the HAZARD margin is $+0.287$.

Candidate	Initial evidence	Additional evidence	Decision
#871	Screening -1.901	An additional seed produces invalid primary predictions.	Confirmation failed
#937	Screening -0.703	Five valid runs; only four improve, and the mean is -0.323.	Confirmation failed
#982	Confirmation passed	Temporal -3.012, spatial -0.268, but HAZARD $+1.531$.	Not accepted
#976	Confirmation passed	Temporal -1.263, spatial -1.683, and HAZARD -0.349.	Accepted

Simple reweighting also improves the parent: fixed weights $(3, 1, 1)$ and $(2, 1, 1)$ achieve development RMSEs of 0.048440 and 0.048870, respectively. Nine confirmed edits have lower development means than the $(3, 1, 1)$ calibration reference. The supplementary confirmation of the $(2, 1, 1)$ reference is reported in Section 5.4.

5.2 Candidate Progression and Verification Failures (RQ2)

Of 70 scored attempts, 24 pass screening: eleven satisfy paired confirmation, twelve fail, and one remains unresolved at the cutoff. All eleven confirmed candidates undergo held-out evaluation, and seven satisfy joint acceptance. Appendix C reports the breakdown by proposal lineage.

Among the 23 screening survivors with resolved confirmation outcomes, twelve fail the confirmation requirements. Table 2 links representative decisions to the additional evidence: invalid primary predictions, inconsistent or insufficient gains, and event-conditioned degradation. Passing an earlier stage establishes eligibility for further evaluation, not the outcome of later stages.

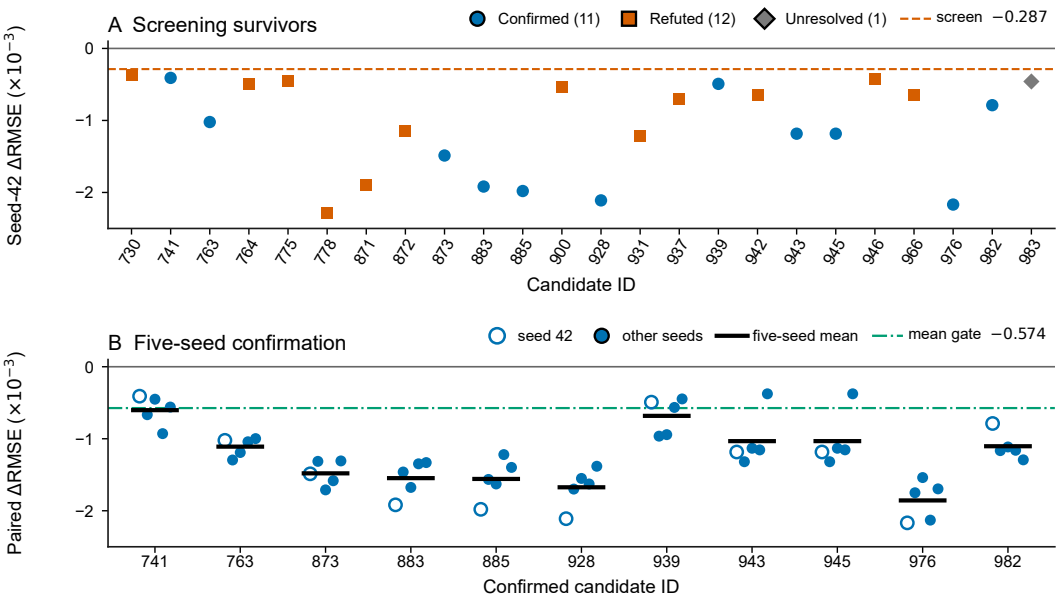


Figure 2: Screening and paired confirmation. (A) The 24 screening survivors, colored by confirmation outcome. (B) Five-seed profiles of the eleven confirmed candidates, with both candidate and parent trained from scratch. Open markers denote the screening seed, filled markers the additional seeds, and black bars five-seed means. Differences are normalized RMSE in units of 10^{-3}; negative is better. The line in (B) marks the mean-effect threshold, one of the joint confirmation requirements; IDs identify archived candidates.

Figure 2 shows the screening and confirmation profiles. Excluding the screening seed leaves four negative differences for every confirmed candidate, with means from -0.001779 to -0.000651, all satisfying the original mean threshold (Appendix E.4).

5.3 Conditional Generalization and Acceptance Sensitivity (RQ3)

All eleven confirmed candidates improve temporal and spatial means, but four exceed the HAZARD mean-degradation margin. The event condition therefore excludes four candidates that satisfy the aggregate holdout and final-validity requirements. The anchor–barrier edit has the largest temporal gain but fails this condition; the adaptive-scale edit achieves a smaller temporal gain while satisfying all three conditions (Table 2). Thus, ranking aggregate gains alone does not identify which candidates satisfy the joint forecasting requirements.

Holding the other requirements fixed, HAZARD margins of 0, m, $2m$, and $4m$ retain five, seven, eight, and ten candidates, respectively. The specified margin admits the two error-reweighting edits in addition to the five candidates with negative HAZARD means: their increases of 0.000208 and 0.000249 remain within the permitted degradation. Appendices B and D report the complete sensitivity analysis and paired-seed intervals.

5.4 Discovered Objectives and Supporting Search Controls (RQ4)

Discovered objective designs. Fourteen of the 19 downstream-objective attempts pass screening, eleven satisfy confirmation, and seven meet final acceptance. Representation, pretraining-objective, and model interventions produce two, three, and five screening survivors, respectively, but no confirmed candidates. Downstream-objective design is the only intervention stream yielding confirmed candidates under this parent and training budget.

The accepted edits span error-dependent reweighting, horizon-conditioned robust penalties, and adaptive residual scaling (Table 18). Some robust variants additionally penalize predictions outside the normalized range.

The adaptive-scale pseudo-Huber edit computes

$$\ell_{hc}(e) = 2d_{hc}^2 \left(\sqrt{1 + (e/d_{hc})^2} - 1 \right), \qquad d_{hc} = 2 \max\big(\mathrm{sg}(r_{hc}), 10^{-3}\big), \tag{5}$$

where e is the residual, r_{hc} is its masked batch RMS for horizon–component cell (h, c), and sg denotes stop-gradient. The loss matches MSE locally around zero and reduces the influence of large residuals relative to the current cell scale. Appendix E.1 gives the exact implementation.

Measured forecasting behavior. The horizon-conditioned robust edit (candidate #883) concentrates development gains at longer horizons, whereas the adaptive-scale edit improves all eighteen development endpoint means with similar gains across horizons. Appendix E reports the detailed profiles and their connections to the implemented objectives.

Search controls. The finite-template reference produces one screening survivor among 28 completed trials. Its weighted-MSE objective yields five valid negative differences and $t_4 = -4.40$, but its mean difference of -0.000486 fails the -0.000574 confirmation threshold.

Without development history, search achieves the better best valid screening result in three of four paired blocks. One block nevertheless exhibits successive within-family refinement with history, reaching -0.001288. This is an example of within-family refinement, not a consistent best-score advantage from history (Appendix F). In the main campaign, candidates #873, #883, and #885 were proposed before candidate #871's confirmation failure became available; that later verdict cannot explain their generation (Appendix E).

5.5 Discussion and Limitations

The four HAZARD rejections show that cross-seed consistency and aggregate held-out gains establish different properties from conditional performance. EvidLoop brings these requirements into a single candidate-level decision process. The discovered reweighting and robust-loss designs provide concrete starting points for component-wise comparisons of their forecasting benefits.

Acceptance follows observed means rather than confidence bounds. All five candidates improving every held-out mean have HAZARD paired-seed intervals containing zero; two (#873 and #885) have upper endpoints below the permitted margin. Conversely, a mean-rule rejection need not establish significant degradation. These pointwise intervals describe training-seed variation on fixed samples, not variation across new weather events. ZSPD and ZSSS each contribute one HAZARD block per horizon and one quarter of the airport-balanced aggregate weight. Evaluation across additional weather periods and parent models would establish how broadly the observed benefits extend.

6 Conclusion

We presented EvidLoop, a framework for agentic model search that links executable edits to staged verification. Development feedback guides proposal generation, paired confirmation tests the same implementation across seeds, and frozen held-out assessment determines whether candidates satisfy joint forecasting requirements. On AeroWF, 70 scored attempts yield eleven confirmed candidates, of which seven satisfy the mean-based held-out acceptance rule. The confirmed edits reduce development RMSE by 1.22%–3.76% relative to the parent, with accepted designs spanning error-dependent reweighting, horizon-conditioned robust penalties, and adaptive residual scaling.

The central empirical finding is that cross-seed improvement and aggregate generalization do not ensure acceptable event-conditioned performance. Four confirmed candidates improve both temporal and spatial means yet exceed the HAZARD degradation margin, including the candidate with the largest temporal gain. By making these distinctions explicit and traceable to evaluated implementations, EvidLoop connects automated search to evidence-based candidate selection. Its contribution is both a set of useful forecasting edits and a traceable research workflow for identifying which candidate improvements survive the specified evaluation conditions.

AI USE STATEMENT

Generative AI was used both within the experimental framework and during research and manuscript preparation. As described in the main text, DeepSeek, GLM, and Kimi generated hypotheses and executable candidate edits from development feedback. Their implementations were evaluated through the execution harness, prediction-validity checks, paired confirmation, and frozen held-out evaluation.

OpenAI's Codex assistant supported manuscript organization and language editing, literature identification, feedback on experimental design, interpretation of results, consistency checks against available experimental records, and figure design and revision. The authors reviewed the AI-assisted material and retain responsibility for experimental decisions, verification, citations, and the final content.

REPRODUCIBILITY STATEMENT

Section 3 specifies the research workflow and acceptance rules, and Section 4 describes their experimental instantiation. Appendix A documents data construction, aggregation, and event support; Appendix B records calibration and policy chronology; and Appendix C provides the frozen candidate accounting and archived source identities. Complete certification summaries and development analyses appear in Appendices D and E. Appendices F and G describe supporting controls, historical training and model-generation settings, artifact provenance, and unavailable fields. These materials support checking the reported candidate decisions and identifying the configurations needed to replay fixed edits. Sealed evaluation remains subject to the custodian's access boundary. Replaying the full adaptive LLM search is distinct from replaying a fixed candidate and is not claimed to reproduce an identical proposal trajectory.

REFERENCES

D. Alves, F. Mendonça, S. S. Mostafa, and F. Morgado-Dias. Low tropospheric wind forecasts in aviation: The potential of deep learning for terminal aerodrome forecast bulletins. *Pure and Applied Geophysics*, 181:2265–2276, 2024. doi: 10.1007/s00024-024-03522-z.

Jonathan T. Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4331–4339, 2019.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305, 2012. URL https://jmlr.org/papers/v13/bergstra12a.html.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, Samira Ebrahimi Kahou, Vincent Michalski, Tal Arbel, Chris Pal, Gael Varoquaux, and Pascal Vincent. Accounting for variance in machine learning benchmarks. In *Proceedings of Machine Learning and Systems*, volume 3, 2021. URL https://proceedings.mlsys.org/paper_files/paper/2021/hash/0184b0cd3cfb185989f858a1d9f5c1eb-Abstract.html.

Gavin C. Cawley and Nicola L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(70):2079–2107, 2010. URL https://jmlr.org/papers/v11/cawley10a.html.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Aleksander Madry, and Lilian Weng. MLE-bench: Evaluating machine learning agents on machine learning engineering. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.07095.

Angelica Chen, David M. Dohan, and David R. So. EvoPrompting: Language models for code-level neural architecture search. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2302.14838.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N. Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. ScienceAgentBench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.05080.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015. doi: 10.1126/science.aaa9375.

Sayash Kapoor and Arvind Narayanan. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4(9):100804, 2023. doi: 10.1016/j.patter.2023.100804.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Xinyu Li, Yunyi Huang, Quan Fang, Yang Yang, Can Zhao, and Kaiquan Cai. AeroWF: A geometric spectral-temporal dual-stream learning framework for aerodrome weather forecasting. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 2, pp. 2814–2825, 2026. doi: 10.1145/3770855.3817822.

Mingquan Liu, Jiangyu Chen, Hanqun Cao, Xujun Zhang, Pengsen Ma, Xiangru Tang, Shuting Jin, Zhuo Yang, Annie Zheng, Tianfan Fu, Fang Wu, and Xiangxiang Zeng. AgentFold: Closed-loop agentic search for protein folding model design. *arXiv preprint arXiv:2608.26747*, 2026. doi: 10.48550/arXiv.2608.26747. Version 2.

Siyi Liu, Chen Gao, and Yong Li. AgentHPO: Large language model agent for hyper-parameter optimization. In *Conference on Parsimony and Learning (CPAL)*, volume 280 of *Proceedings of Machine Learning Research*, pp. 1146–1169. PMLR, 2025. arXiv:2402.01881.

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

Preetum Nakkiran and Jarosław Błasiok. The generic holdout: Preventing false-discoveries in adaptive data science. *arXiv preprint arXiv:1809.05596*, 2018. URL https://arxiv.org/abs/1809.05596.

Jingjie Ning, Xiaochuan Li, Ji Zeng, Chenyan Xiong, and Guolin Ke. Closed-loop auto research for molecular property prediction: Discovering and certifying generalizable improvements. *arXiv preprint arXiv:2606.22731*, 2026a. Version 1.

Jingjie Ning, Xiaochuan Li, Shanshan Zhong, Ji Zeng, and Guolin Ke. Auto research for materials: Auditable AI-scientist workflows with held-out transfer. *arXiv preprint arXiv:2607.17100*, 2026b. doi: 10.48550/arXiv.2607.17100. Version 2.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. AlphaEvolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

Luke Oakden-Rayner, Jared Dunnmon, Gustavo Carneiro, and Christopher Ré. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 151–159. Association for Computing Machinery, 2020. doi: 10.1145/3368555.3384468.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024. doi: 10.1038/s41586-023-06924-6.

Samuel Schmidgall and Michael Moor. AgentRxiv: Towards collaborative autonomous research. *arXiv preprint arXiv:2503.18102*, 2025.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Michael Moor, Zicheng Liu, and Emad Barsoum. Agent laboratory: Using LLM agents as research assistants. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 5977–6043, Suzhou, China, 2025. Association for Computational Linguistics. arXiv:2501.04227.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Chenglei Si, Tatsunori Hashimoto, and Diyi Yang. The ideation-execution gap: Execution outcomes of LLM-generated versus human research ideas. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Fllp8l6Puy.

Hjalmar Wijk, Tao Roa Lin, Joel Becker, Sami Jawhar, Neev Parikh, Thomas Broadley, Lawrence Chan, Michael Chen, Joshua M. Clymer, Jai Dhyani, Elena Ericheva, Katharyn Garcia, Brian Goodrich, Nikola Jurkovic, Megan Kinniment, Aron Lajko, Seraphina Nix, Lucas Jun Koba Sato, William Saunders, Maksym Taran, Ben West, and Elizabeth Barnes. RE-bench: Evaluating frontier AI r&d capabilities of language model agents against human experts. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 66772–66832. PMLR, 2025. arXiv:2411.15114.

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

Jiakang Yuan, Xiangchao Yan, Bo Zhang, Tao Chen, Botian Shi, Wanli Ouyang, Yu Qiao, Lei Bai, and Bowen Zhou. Dolphin: Moving towards closed-loop auto-research through thinking, practice, and feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21768–21789, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1056. arXiv:2501.03916.

A Data and Evaluation Contract

A.1 Frozen Data Materialization

The downstream AeroWF release contains 108,404 input windows from four airports, with 27,101 original windows per airport. ZBAA, ZSPD, and ZSSS provide training, validation, and temporal holdout data. ZBAD is excluded from model-side development and reserved for spatial evaluation.

The frozen materialization removes 104 preceding-split windows at each of six training–validation or validation–temporal boundaries across the three development airports. This removes 624 windows and leaves 107,780 retained windows. Table 3 reports the allocation.

A.2 Input Windows and Forecast Targets

Each input contains 96 observations at one-minute internal cadence. Adjacent input-window starts are separated by 15 minutes. The runway dimension is padded to four positions, with a real-runway mask identifying the positions corresponding to observed runways.

Table 3: Downstream window accounting. The purged windows are excluded from all retained partitions. Window counts do not represent independent weather events or independent forecast targets.

Partition	Windows
Training	56,598
Validation	11,883
Sealed temporal	12,198
Sealed spatial (ZBAD)	27,101
Total retained	107,780
Purged boundary windows	624
Original release	108,404

Forecast targets are the final-timestep horizontal wind components of same-airport, same-split partner windows at T+1, T+4, and T+8. These offsets correspond to 15, 60, and 120 minutes after the final observation of the input window. For example, an input beginning at time t ends at $t + 95$ minutes; its three target times are $t + 110$, $t + 155$, and $t + 215$ minutes.

A missing partner produces an unavailable target. For training and validation loss computation, `AirportForecastDataset` uses `require_complete_horizons=True`: an anchor is retained only when all three same-airport, same-split partners at T+1, T+4, and T+8 are available. This excludes the last eight anchors per airport, whose T+8 partners fall outside the partition; anchors missing shorter-horizon partners are a subset of these rows.

Consequently, the 56,598 training source windows yield 56,574 forecasting loss samples (18,858 per airport). The 11,883 validation source windows yield 11,859 validation loss samples (3,953 per airport). Prediction export retains anchors with incomplete horizon coverage, and the evaluator excludes unavailable targets separately at each endpoint. Thus, loss-sample filtering does not reduce the exported source-window coverage.

Table 4: Source-window and forecasting loss-sample counts across the three development airports. Complete-horizon filtering excludes eight tail anchors per airport from loss computation. Prediction export retains these anchors.

Partition	Source windows	Loss samples	Difference
Training	56,598	56,574	24
Validation	11,883	11,859	24

A.3 Preprocessing and Temporal Provenance

Leakage in data preparation can compromise empirical findings in machine-learning-based science (Kapoor & Narayanan, 2023). We therefore document preprocessing scope separately from the timestamp provenance of the archived observations.

Normalization parameters are fitted only on retained training samples from ZBAA, ZSPD, and ZSSS. Validation, temporal holdout, and ZBAD samples are excluded from normalization fitting. The evaluator applies the prescribed scored-position range checks to predictions; invalid outputs are not silently clipped or repaired.

METAR reconstruction follows the archive's observation/report-time alignment. Historical issue and receive timestamps are unavailable, so report age provides a lower-bound proxy for operational staleness. The available timestamps support the documented observation-time construction but cannot establish when every report became available at the forecast origin.

The development workflow does not expose sealed inputs, labels, private masks, or sealed event membership to the proposal agents. The custodian evaluates the frozen candidates and provides the approved aggregate readouts.

A.4 Endpoint Aggregation

For each seed, the evaluator first computes wind RMSE within every airport–horizon–component endpoint. Valid scoring positions are pooled within an endpoint; the resulting endpoint RMSE values then receive equal weight.

The development and temporal readouts each contain 18 endpoints: three airports, three forecast horizons, and two horizontal wind components. The spatial readout contains six endpoints from ZBAD. The HAZARD-conditioned readout contains 24 endpoints across all four airports. Endpoint weights do not depend on their numbers of scoring positions.

For each readout, the candidate's aggregate score is compared with its prescribed same-seed parent score. The five resulting paired differences receive equal weight in the reported mean. This aggregation differs from computing one RMSE over all pooled airports and horizons.

Table 5 reports the valid scoring positions in the approved certification summary. Each count applies separately to either wind component. These positions can share input windows, forecast targets, and weather episodes.

Table 5: Valid scoring positions per airport–horizon–wind-component endpoint. Each count applies separately to each wind component and is not a count of independent windows, dates, or weather events.

Readout	Airport	T+1	T+4	T+8
Temporal	ZBAA	12,195	12,186	12,174
Temporal	ZSPD	16,260	16,248	16,232
Temporal	ZSSS	8,130	8,124	8,116
Spatial	ZBAD	108,400	108,388	108,372
HAZARD	ZBAA	1,350	1,350	1,350
HAZARD	ZSPD	372	372	372
HAZARD	ZSSS	66	66	66
HAZARD	ZBAD	3,160	3,160	3,160

A.5 HAZARD Definition and Sealed Event Support

The frozen HAZARD ontology selects target-time partner windows whose `exo_cat_weather_code_id` belongs to

$$\{3, 4, 5, 14, 15, 16, 17, 18, 19, 20\}. \tag{6}$$

The category rule is applied separately at each forecast horizon. A gust flag alone is insufficient for inclusion, and PRECIP category IDs 6–13 do not enter the HAZARD slice solely through their category membership.

The HAZARD readout applies this mask to the temporal and spatial predictions. It evaluates conditional wind-forecasting performance; weather-classification accuracy does not determine candidate acceptance.

Event-block construction. The custodian supplied aggregate event-block and UTC-date counts for the sealed HAZARD slice (Table 6). The summary was produced without rerunning models or changing the original certification verdicts.

Within each airport–horizon pair, the summary counts available HAZARD targets with valid event-block membership. Targets are ordered by target time. Consecutive targets separated by no more than 90 minutes are joined into the same block. Overlapping windows are not treated as separate events. UTC-date counts refer to distinct target dates within the corresponding airport–horizon pair.

At each horizon, the counts sum to 17 airport-specific blocks and 32 airport–date incidences. These totals do not establish the number of unique meteorological events across airports or the number of distinct dates across the full sealed slice. The same weather event may also contribute at multiple horizons, so summing the three horizon counts would not give a unique event count.

Table 6: Custodian-reported support for the sealed HAZARD readout. Each entry gives event blocks / distinct UTC target dates. Blocks are constructed separately by airport and horizon; cross-horizon counts are not deduplicated.

Airport	T+1	T+4	T+8
ZBAA	3 / 8	3 / 8	3 / 8
ZSPD	1 / 2	1 / 2	1 / 2
ZSSS	1 / 2	1 / 2	1 / 2
ZBAD	12 / 20	12 / 20	12 / 20

Each airport contributes six of the 24 HAZARD endpoints and therefore receives one quarter of the aggregate weight. The event support is uneven: ZSPD and ZSSS each contribute one block, whereas ZBAD contributes twelve. These counts describe the event concentration underlying the airport-balanced readout. The reported paired-seed intervals characterize variation across training seeds on these fixed samples; they do not quantify uncertainty across new weather events.

A.6 Training-Side HAZARD Support

Training-side event support is obtained separately from the complete 56,598-row O2 training flags. Using the within-airport, within-horizon rule that joins consecutive event targets no more than 90 minutes apart, T+1 contains 734 HAZARD targets in eight blocks. T+4 and T+8 have the same aggregate counts. Table 7 reports the airport-level support.

Table 7: Training-side HAZARD support at T+1. T+4 and T+8 have the same aggregate counts. Airport-days count a date separately for each airport. These counts concern the training flags and are separate from the sealed certification support.

Airport	HAZARD targets	Event blocks	UTC dates
ZBAA	296	3	6
ZSPD	324	4	6
ZSSS	114	1	2
Total	734	8	14 airport-days

Across the three training airports, these targets occur on nine distinct UTC calendar dates. The training-side block and date counts describe the support available for training interventions. The sealed support in Table 6 describes the separate samples used for final conditional evaluation.

B Promotion Policy, Calibration, and Chronology

B.1 Paired Effects and Calibration

Candidate effects are measured against the prescribed same-seed parent reference:

$$\delta^e_{i,s} = M_e(f_{i,s}) - M_e(f_{0,s}). \tag{7}$$

Negative differences favor the candidate. The primary comparison leg is retained in the candidate record. All confirmed candidates use scratch-trained candidate and parent models.

The campaign's effect scale comes from a fixed horizon-weighted MSE probe with weights $(3, 1, 1)$. The probe configuration and five-seed evaluation were specified before the first recorded Regime B candidate result. Table 8 gives the recorded paired differences.

The standard deviation supplies the operational scale $m = 0.000287$. The campaign uses one such standard deviation for screening and two for the confirmation mean-effect floor. Screening allocates additional evaluation to promising edits; confirmation requires a larger average effect across the prescribed seeds.

The probe itself has a favorable, nonzero mean effect. Its standard deviation therefore measures variation in the paired effect of this particular intervention, including seed-dependent behavior of

Table 8: Calibration-probe differences against the same-seed scratch parent. The configuration uses horizon-weighted MSE with weights $(3, 1, 1)$. Values are normalized-RMSE differences in their original units. Seed-level entries are rounded for display; the mean is computed from the full-precision scores.

Seed	Probe minus parent
42	-0.001133
43	-0.000723
2027	-0.001202
3407	-0.001000
5519	-0.000521
Mean	-0.000915931
Sample standard deviation	0.000287

the edit. It is not an estimate from a zero-effect control. The multipliers are campaign policy choices rather than optimized thresholds or aviation-specific tolerances.

The calibration records, IDs 520–524, are excluded from both the 70 scored LLM attempts and the 28 completed random-template trials. Their weights also differ from those of `rand-obj-611`, which uses $(2, 1, 1)$. They do not provide confirmation evidence for that random-template candidate.

B.2 Screening and Confirmation Rules

Screening. The screening seed is 42. A candidate passes when its prescribed primary readout is valid and

$$\delta^{\mathrm{dev}}_{i,42} \le -m = -0.000287. \tag{8}$$

The decision uses the candidate's registered primary comparison field. Auxiliary readouts are retained as diagnostics and do not replace that field after scores are observed.

A screening pass records eligibility for further evaluation. Completion of confirmation is a separate recorded outcome. Candidate 983 remained unresolved at the campaign cutoff, and the random-template survivor 611 was not entered into the original confirmation queue. A later supplementary confirmation of 611 is recorded in Appendix C.5.

Paired confirmation. Confirmation uses the same executed edit on

$$\mathcal{S} = \{42, 43, 2027, 3407, 5519\}. \tag{9}$$

The screening result supplies the seed-42 observation; the other four seeds provide additional paired results. For five valid primary differences, define

$$\bar{\delta}_i = \frac{1}{5}\sum_{s\in\mathcal{S}} \delta^{\mathrm{dev}}_{i,s}, \qquad s_{\delta,i} = \sqrt{\frac{1}{4}\sum_{s\in\mathcal{S}} \left(\delta^{\mathrm{dev}}_{i,s} - \bar{\delta}_i\right)^2}. \tag{10}$$

A candidate is confirmed only if

$$\max_{s\in\mathcal{S}} \delta^{\mathrm{dev}}_{i,s} < 0, \qquad \bar{\delta}_i \le -0.000574, \qquad \frac{\bar{\delta}_i}{s_{\delta,i}/\sqrt{5}} \le -2.776, \tag{11}$$

with all five required primary readouts valid.

The statistic threshold is the two-sided 5% t critical value with four degrees of freedom, applied in the required negative direction. Its conjunction with the sign and mean-effect conditions defines the promotion rule. Because confirmation includes the screening seed and occurs within adaptive development, these conditions are not interpreted as independent tests or as family-wise error control for the search.

Early stopping and incomplete evidence. Confirmation can terminate once a required validity or sign failure makes five valid negative effects unattainable. The observed results and stopping reason

remain in the record. An invalid primary observation has no usable numerical difference and is not replaced by zero. A partial mean is not reported as a complete five-seed confirmation result.

A failed confirmation receives a *refuted* promotion verdict. An unfinished confirmation remains *unresolved*. These promotion states are distinct from trial-level execution labels. For example, a trial-level invalid label can concern an auxiliary output while the primary forecasting readout remains valid.

B.3 Final Acceptance Rule

All eleven confirmed candidates were included in the frozen set submitted for sealed evaluation. For each candidate, the custodian evaluated the five seed-specific models against their corresponding parent references. Final acceptance requires valid primary predictions and the following mean paired effects:

$$\bar{\delta}_i^{\text{temporal}} < 0, \qquad \bar{\delta}_i^{\text{spatial}} \leq +0.000287, \qquad \bar{\delta}_i^{\text{HAZARD}} \leq +0.000287. \tag{12}$$

The spatial and HAZARD conditions permit bounded mean degradation using the same operational scale m as the development policy. A candidate satisfying all conditions is *transferred*. A completed final assessment that fails a required condition leaves the candidate *validation-only*. An incomplete final assessment would remain *inconclusive*; all eleven confirmed candidates in the reported campaign received completed final assessments.

These are mean-based acceptance conditions. Confidence intervals provide additional information about the observed effects but do not replace the registered thresholds. In particular, transferred status does not require a confidence bound to lie below the corresponding margin.

B.4 Policy Provenance and Freeze Chronology

The development and certification rules have distinct registration histories. Policy v1.4 and the paired-probe calibration established the development screening and confirmation criteria before the first recorded Regime B candidate result. Certification conditions were documented after development search had begun and before sealed access.

Table 9 summarizes the available records. The first visible Regime B completion is `llm-rep-700` at 2026-09-06 10:52 UTC. Policy recalibration is recorded at approximately 15:5x China Standard Time on the same date, before that completion. A completed-trial timestamp does not establish the time of the first proposal, so the supported description is "before the first recorded candidate result."

Table 9: Chronology of the development policy, certification rules, and sealed evaluation. All dates refer to September 2026. The certification conditions were fixed during discovery and before sealed access.

Component	Record date	Supported timing
Development policy v1.4	September 5–6	Screening and confirmation criteria documented before the first recorded Regime B result.
Probe calibration	September 6	$m = 0.000287$ recorded before the first Regime B completion.
Temporal and spatial conditions	September 6 handoff; September 7–8 plan	Documented during discovery and before sealed evaluation.
Certification margin	September 8	Custodian signed the $+m$ margin before candidate freeze and sealed access.
Candidate-set freeze	September 13	Confirmed set fixed before sealed evaluation.
Sealed evaluation	September 14–15	One-time access after candidate freeze and rule registration.

The available records do not establish a pre-search freeze of the temporal, spatial, and HAZARD acceptance conditions. They establish that these conditions preceded the sealed readouts to which they were applied. Final outcomes assigned candidate statuses and did not feed further proposals within the campaign.

Source digests identify archived artifacts and allow their consistency to be checked. The chronology additionally relies on the dated policy, handoff, signing, and access records; a digest alone does not establish when a rule was fixed.

B.5 Descriptive Sensitivity to the Event Margin

We examine the effect of the HAZARD margin using the same eleven frozen certification profiles. The temporal, spatial, and validity conditions remain fixed while only the permitted HAZARD mean degradation changes. Table 10 reports the resulting candidate counts.

Table 10: Post hoc sensitivity to the HAZARD mean-degradation margin. All other final requirements remain fixed. The registered campaign margin is $m = 0.000287$.

Margin	Numerical value	Candidates retained
0	0	5
m	0.000287	7
$2m$	0.000574	8
$4m$	0.001148	10

With a zero margin, only the five candidates with negative HAZARD means remain. The registered margin additionally accepts 741 and 763. Doubling the margin admits 939, and a margin of $4m$ also admits 943 and 945. Candidate 982 remains outside the accepted set at every displayed margin.

This analysis describes how the tolerance affects decisions on the frozen profiles. It does not select a replacement margin or revise the original seven transferred and four validation-only outcomes.

B.6 Campaign Execution Procedure

The campaign links proposal generation, candidate evaluation, and final assessment through the following procedure:

1. Specify the parent reference, permitted editing surfaces, training budget, evaluator, and comparison contracts. Record the effective timing and provenance of the stage-specific decision rules.
2. Assemble each proposal context from campaign facts and development observations available to that lineage at proposal time. Later outcomes are not treated as evidence available to an earlier proposal.
3. Generate a structured hypothesis and patch plan, then produce the executable edit. Check file boundaries, interfaces, and smoke-test behavior before training.
4. Execute the screening experiment with seed 42. Record the executed source identity, required primary validity, prescribed paired comparison, auxiliary diagnostics, and execution status.
5. Evaluate admitted screening survivors on the additional confirmation seeds using the same executed edit and the corresponding same-seed parent references. Retain completed observations, early-stopping reasons, and unresolved outcomes.
6. Make newly available development results accessible to subsequent proposals through the lineage. Historical edits inform proposals without replacing the prescribed parent comparison.
7. At the campaign cutoff, freeze the confirmed set with source identities, seed-specific checkpoints, parent references, and inference specifications. Preserve the accounting of non-promoted records.
8. Have the custodian evaluate every frozen confirmed candidate once on the temporal and spatial holdouts and their HAZARD-conditioned slice. Apply the joint acceptance conditions fixed before sealed evaluation.
9. Report the complete final profiles and candidate statuses. Do not use sealed readouts to generate further edits or revise the acceptance criteria within the frozen campaign.

C Candidate Accounting and Frozen Ledger

C.1 Reporting Units and Status Fields

The frozen campaign distinguishes execution records, scored candidate records, and subsequent verification outcomes. Its 80 execution records comprise 57 completed records, 13 scored-invalid records, and ten execution failures. The completed and scored-invalid records form the 70-attempt screening denominator. Additional confirmation seeds contribute evidence to an existing candidate rather than new proposals.

Table 11: Candidate progression by proposal lineage. Counts refer to candidate records; accepted records have the historical status `transferred`. The lineages have unequal search exposure.

Lineage	Scored	Screen-pass	Confirmed	Accepted
DeepSeek	26	6	2	2
GLM	24	8	4	4
Kimi	20	10	5	1
Total	70	24	11	7

Execution status and promotion status are retained separately. A trial-level `invalid` label can concern an auxiliary output while the primary forecasting readout remains valid. Consequently, trial-level invalidity and candidate-level confirmation can coexist. An invalid primary readout, in contrast, cannot supply an admissible improvement for promotion.

The development outcomes are:

- *Screen-fail*: the record does not satisfy the applicable screening requirements.
- *Refuted*: a screening survivor fails the confirmation rule.
- *Confirmed*: the candidate satisfies the complete five-seed confirmation rule.
- *Unresolved*: a screening survivor has no completed confirmation verdict at the cutoff.

Sealed evaluation subsequently assigns transferred or validation-only status to confirmed candidates.

Table 12 reconciles execution and promotion counts by intervention stream. Withdrawn candidate 782 did not enter the scored census. The identifier `llm-rep-930` appears in both a failed execution record and a completed scored record. These remain separate execution records; only the completed record enters the scored denominator. Thus, execution totals should not be interpreted as counts of unique candidate identifiers.

Table 12: Frozen accounting by intervention stream. Executed counts include failures. Scored counts include completed and scored-invalid records. Confirmation and transfer counts refer to candidate records.

Stream	Executed	Scored	Screened	Confirmed	Transferred
Representation	30	28	2	0	0
Downstream objective	19	19	14	11	7
Pretraining objective	7	7	3	0	0
Model	24	16	5	0	0
Total	80	70	24	11	7

C.2 Decision Fields and Source Provenance

The candidate index joins each scored record to its frozen seed-42 decision value and archived edit digest. Decision values follow the field stored for the individual record. For screening survivors, `campaign.screen_primary` resolves to the scratch comparison for representation and downstream-objective candidates and to the registered tier-2 comparison for the relevant pretraining-objective and model candidates. Other records retain their stored scratch, tier-2, or unavailable value. The comparison leg is therefore not assigned from the model-edit label alone.

For example, candidate 983 passes screening with a tier-2 difference of -0.000459. Its separate scratch value of zero is not the effect used for that screening decision. A numerical difference also does not establish eligibility without the applicable validity checks: candidate 906 retains a difference below the numerical screening threshold but has a screen-fail outcome.

All 70 edit digests are available in the `artifact_digests.edit_sha256` field of the September 13 A-side frozen trial `result.json` files. The audited digests match the corresponding D, OT1, and R trial metadata for all 70 records. They are archived metadata, not hashes reconstructed from current source files. The original lineage rows do not themselves contain these source-digest fields.

Table 13 displays the first twelve hexadecimal characters of each digest. These prefixes are unique within the 70-record index. The accompanying `scored_index_70.csv` retains the full 64-character digests, host identifiers, decision-field provenance, and numerical values at their archived precision.

C.3 Complete Scored-Candidate Index

Table 13 reports all scored LLM candidate records. Stream abbreviations are R for representation, O1 for downstream objective, O2 for pretraining objective, and M for model. Candidate numbers are the suffixes of the archived trial identifiers: `llm-rep-` for R, `llm-obj-` for O1/O2, and `llm-model-` for M.

A dash denotes an unavailable decision value. A displayed 0.000 denotes a recorded zero. In particular, the eight representation zeros are observations rather than missing values. The development-status column stops at confirmation; final statuses for the confirmed set are reported in the certification appendix.

Table 13: All 70 scored LLM records at the September 13 freeze. Seed-42 differences are in units of 10^{-3}. Trial status and development outcome are separate fields. SHA prefixes identify archived edit digests.

ID	Stream	Leg	Trial status	Seed-42 Δ	Development outcome	SHA prefix
700	R	scratch	completed	0.000	screen-fail	6107d9405696
726	R	scratch	completed	+0.481	screen-fail	84775f089691
728	R	scratch	completed	+0.618	screen-fail	a2003b773d7c
729	O1	scratch	completed	+0.010	screen-fail	3a892fa46c75
730	O2	t2	completed	−0.371	refuted	1bc37fb17c6a
732	R	scratch	completed	+4.077	screen-fail	92a9339835e0
733	O1	scratch	completed	+2.119	screen-fail	ce4a6073a11f
734	R	scratch	completed	0.000	screen-fail	98f0d451035f
735	O1	scratch	completed	−0.087	screen-fail	8faf5092d8f8
738	R	scratch	completed	+5.339	screen-fail	3a4f21cbf5b6
739	O1	scratch	completed	+0.453	screen-fail	b2b4ff755f74
740	R	scratch	completed	−0.073	screen-fail	f8f61de4ce20
741	O1	scratch	completed	−0.409	confirmed	1a362a8e6c42
742	O2	t2	completed	−0.258	screen-fail	fff3703ae372
743	M	scratch	completed	−0.017	screen-fail	a8c4979c7b23
744	M	scratch	completed	+0.070	screen-fail	2a1fd7071f17
762	M	scratch	invalid	+0.421	screen-fail	d92f2bb8bb6a
763	O1	scratch	completed	−1.022	confirmed	5b49ad8a716d
764	M	t2	completed	−0.493	refuted	7b79ec9cb90f
774	R	scratch	completed	0.000	screen-fail	d48feb70d455
775	M	t2	completed	−0.450	refuted	ff18dae3a79a
776	R	scratch	completed	+0.400	screen-fail	4689f0a2c52e
777	M	scratch	completed	+0.294	screen-fail	3c0fc2414f3d
778	O1	scratch	completed	−2.282	refuted	fa93235be82b
779	O2	t2	completed	+1.690	screen-fail	26b8ecec9d24
781	M	scratch	completed	−0.016	screen-fail	ae6c8197f256
820	R	scratch	completed	0.000	screen-fail	7074b64aabf7
852	R	–	invalid	–	screen-fail	fba0b7aec0eb
870	R	scratch	completed	0.000	screen-fail	2f007832864b
871	O1	scratch	completed	−1.901	refuted	2248711423f2
872	R	scratch	completed	−1.144	refuted	6cd5a356628c
873	O1	scratch	invalid	−1.487	confirmed	7ddec2b68d57
874	O2	t2	completed	+1.979	screen-fail	8767935d601c
882	R	scratch	completed	+0.341	screen-fail	ce1080e5323b
883	O1	scratch	invalid	−1.918	confirmed	418b3858b464
884	R	scratch	completed	+0.025	screen-fail	e33f775dfd60
885	O1	scratch	invalid	−1.980	confirmed	e04347209465
888	R	scratch	completed	+0.008	screen-fail	14a986f66a74

ID	Stream	Leg	Trial status	Seed-42 Δ	Development outcome	SHA prefix
889	O1	scratch	completed	−0.044	screen-fail	07381f59cfe5
890	R	scratch	completed	0.000	screen-fail	349dec5ee2c6
900	M	t2	invalid	−0.536	refuted	c0e1fe928b16
906	M	scratch	invalid	−0.518	screen-fail	666e31e08350
910	R	scratch	completed	+0.336	screen-fail	0b9e816dfa2b
918	R	scratch	completed	+0.976	screen-fail	6cbb17dcfd9b
924	R	scratch	completed	−0.115	screen-fail	e861c0a6749d
926	R	scratch	completed	+0.492	screen-fail	9f03a37ae223
927	M	–	invalid	–	screen-fail	ccf785bcc885
928	O1	scratch	completed	−2.110	confirmed	8b99afddf26f
930	R	scratch	completed	+3.551	screen-fail	02ba513d817a
931	M	t2	completed	−1.211	refuted	06505aeb3bab
932	R	–	invalid	–	screen-fail	ad5eef7931c9
936	R	scratch	completed	−0.063	screen-fail	9a166a98cdbc
937	O1	scratch	completed	−0.703	refuted	36e94a2f3f72
939	O1	scratch	completed	−0.492	confirmed	731cb5ba46a5
942	R	scratch	completed	−0.647	refuted	ed79a34b0a08
943	O1	scratch	invalid	−1.184	confirmed	31600b332134
944	R	scratch	completed	0.000	screen-fail	35ca57329641
945	O1	scratch	invalid	−1.184	confirmed	5aeabc53cfd3
946	O2	t2	completed	−0.418	refuted	f75c680618af
948	M	scratch	completed	−0.026	screen-fail	3a4dd9e639f8
960	M	scratch	completed	+0.014	screen-fail	5db4717f104f
966	M	t2	completed	−0.649	refuted	7ceaf53627c0
974	R	scratch	completed	0.000	screen-fail	b9f0ee4029c0
975	M	scratch	completed	+0.663	screen-fail	d98d851ee755
976	O1	scratch	completed	−2.168	confirmed	4ecc361f9eae
977	O2	t2	invalid	–	screen-fail	107c97c9a2f3
980	R	scratch	completed	+0.751	screen-fail	336d950f7506
981	M	–	invalid	–	screen-fail	43489241f22b
982	O1	scratch	completed	−0.788	confirmed	1af9375cbd43
983	O2	t2	completed	−0.459	unresolved	9b6db77e9868

C.4 Screening Survivors and Confirmation Dispositions

The 24 screening survivors comprise eleven confirmed candidates, twelve refuted candidates, and one unresolved candidate. Table 14 links each survivor to its proposal-family description and development verdict. The descriptions summarize the proposed or implemented intervention and do not imply a validated causal mechanism.

Table 14: All screening survivors and their frozen confirmation dispositions. Screening differences are in units of 10^{-3}.

ID	Intervention description	Screen Δ	Leg	Verdict
730	T+1 reconstruction upweighting	−0.371	t2	Refuted
741	Inverse-error horizon balancing	−0.409	scratch	Confirmed
763	Horizon–component balancing	−1.022	scratch	Confirmed
764	Wind-delta final-token damping	−0.493	t2	Refuted
775	Wind-spectrum tempering	−0.450	t2	Refuted
778	Horizon-scaled Charbonnier	−2.282	scratch	Refuted
871	Pseudo-Huber tail robustness	−1.901	scratch	Refuted
872	Recency input gain	−1.144	scratch	Refuted
873	Horizon-conditioned robust curvature	−1.487	scratch	Confirmed
883	Horizon-conditioned Huber elasticity	−1.918	scratch	Confirmed
885	Horizon-conditioned pseudo-Huber	−1.980	scratch	Confirmed
900	Per-channel frequency gate	−0.536	t2	Refuted
928	Boundary-anchored robust loss	−2.110	scratch	Confirmed
931	Trend-to-present injection	−1.211	t2	Refuted
937	Tendency-coupled objective	−0.703	scratch	Refuted
939	Adaptive cell weighting	−0.492	scratch	Confirmed
942	Causal tendency tilt	−0.647	scratch	Refuted
943	Anchored wind-range barrier	−1.184	scratch	Confirmed
945	Anchored range-hinge barrier	−1.184	scratch	Confirmed
946	Causal reconstruction emphasis	−0.418	t2	Refuted

ID	Intervention description	Screen Δ	Leg	Verdict
966	Adaptive range-limit fusion	-0.649	t2	Refuted
976	Adaptive-scale bounded influence	-2.168	scratch	Confirmed
982	Robust anchor–barrier fusion	-0.788	scratch	Confirmed
983	Pretraining-objective edit	-0.459	t2	Unresolved

Candidates 778 and 871 illustrate primary-validity failures during confirmation. Candidate 778 has four favorable valid differences, but the invalid seed-43 primary readout prevents five valid negative results. Candidate 871 likewise fails after primary invalidity during replay. Candidate 937 instead completes five valid runs, with only four negative differences and a mean of -0.0003232, above the required effect floor. These outcomes distinguish unavailable evidence from unfavorable valid effects.

Candidate 983 passes its tier-2 screen but has no completed confirmation at the freeze. It is retained as unresolved rather than counted among either confirmed or refuted candidates. No candidate screened under the pretrained-to-scratch comparison satisfies the confirmation rule in the reported campaign.

C.5 Shared Finite-Template Reference

The random-template arm supplies one shared reference for all three LLM proposal lineages. Its 28 completed screening trials are comparable in count to the 26, 24, and 20 scored attempts from DeepSeek, GLM, and Kimi, respectively. These counts describe the scale of the executed searches: the random denominator includes completed trials, whereas the LLM denominators also include scored-invalid records. The shared reference is counted once, rather than treated as three independent random-search replications. Its generator contains six template families: three downstream-objective templates and three representation templates. Their parameter lists contain 36 literal combinations in total, forming a union of template-specific settings. The generator includes no model or pretraining-objective intervention.

The frozen reference comprises ten original controls and eighteen completed clone trials (Table 15). The clone segment uses new draws with sampling seed 20260910 and sequence numbers starting at 600. A draw matching an already evaluated configuration reuses that result without creating a new GPU trial. The eighteen completed canonical clone records represent distinct configurations.

Table 15: Membership of the 28 completed random-template records. Numbers are trial-identifier suffixes. The reference is shared once across the three LLM lineages.

Source	Count	Trial suffixes
Original D controls	3	530, 532, 533
Original OT1 controls	4	550, 551, 553, 554
Original R controls	3	571, 572, 574
Completed clone trials	18	604, 605, 608, 611, 614, 615, 617, 621, 627, 628, 629, 630, 635, 638, 640, 650, 658, 659
Total	28	

The original ten controls produced no eligible screening survivor. The additional sampling was authorized after that outcome was known. The planned top-up yielded eighteen completed trials rather than twenty, with repeated configurations ending further supplementation. This stopping history does not establish exhaustion of all 36 literal configurations.

Calibration probes 520–524, failed clone trial 600, and scored-invalid clone trials 601, 603, and 607 are outside the completed-trial denominator. Including the latter three scored-invalid records would produce a denominator of 31 while leaving one eligible screening survivor. The published 28-record reference therefore uses a different validity-accounting denominator from the 70 scored LLM attempts.

The completed random trials share the v2p3 parent, seed-42 evaluation, evaluator, and downstream training budget with the LLM campaign. The proposal processes differ: the random arm instantiates

static templates, whereas the LLM workflow permits iterative generation and repair. The reference consequently characterizes fixed-template output under the common execution pipeline within its covered intervention space.

The screening survivor. Candidate `rand-obj-611` uses horizon-weighted MSE with weights $(2, 1, 1)$ and records a seed-42 scratch difference of -0.000616. It is the sole eligible screening survivor in the completed random reference.

The original September 13 freeze contains only the seed-42 screening result for this candidate. The random arm served as a screening reference and was outside the original LLM confirmation queue.

Post-freeze confirmation of `rand-obj-611`. A supplementary evaluation completed the five-seed scratch–scratch comparison under the original v1.4 confirmation criteria. The seed-42 screening run completed on September 11, 2026; the additional confirmation runs completed on September 24–25. The final `confirmation_failed` verdict was recorded on September 25, 2026, at 02:43 UTC in the clone host's `lineage.jsonl` and `lineage_canonical.jsonl`.

The execution records include `confirm_rand-obj-611.log`, `confirm_rand-obj-611.sh`, and `disc_rand_obj_611_seed42.stdout`. This supplementary evaluation extends the candidate's evidence record without changing the September 13 campaign census.

Table 16: Supplementary confirmation of `rand-obj-611`. Differences are scratch–scratch paired normalized RMSE in units of 10^{-3}. All five primary readouts are valid. Displayed differences are rounded.

Seed	Primary Δ	Trial-level status
42	-0.616	Completed (screening)
43	-0.516	Completed
2027	-0.758	Completed
3407	-0.443	Invalid; scratch primary valid
5519	-0.098	Completed

The five differences are negative and the recorded paired statistic passes the required threshold ($t_4 = -4.40 \leq -2.776$). The mean difference is approximately -0.486 in the displayed units, which does not satisfy the confirmation floor of -0.574. The candidate is therefore refuted on the mean-effect criterion, not on sign consistency or primary prediction validity.

The seed-3407 trial has an invalid pretrained output, but its scratch primary readout remains valid and contributes to the confirmation decision. Discarding that paired difference solely because of the trial-level status would conflate auxiliary output validity with primary eligibility. The calibration probe with weights $(3, 1, 1)$ remains a different configuration and supplies none of these confirmation observations.

D Certification Readout and Uncertainty

D.1 Complete Sealed Evaluation

All eleven confirmed candidates entered the one-time sealed evaluation. Five valid paired seeds were available for every candidate and readout. Temporal evaluation covers future periods at the three development airports, spatial evaluation covers ZBAD, and the HAZARD-conditioned readout applies the frozen target-time mask to predictions from both holdouts. The conditional readout requires no additional model training or inference.

For each candidate and readout, the custodian's aggregate summary provides the mean paired effect and its sample standard deviation across the five training seeds. Table 17 reports these values together with the corresponding paired-seed intervals. The displayed status is the candidate's joint acceptance outcome, repeated across its three readout rows.

Cumulative acceptance conditions. Applying the final requirements cumulatively to the eleven confirmed records retains eleven after temporal mean improvement, eleven after the spatial margin,

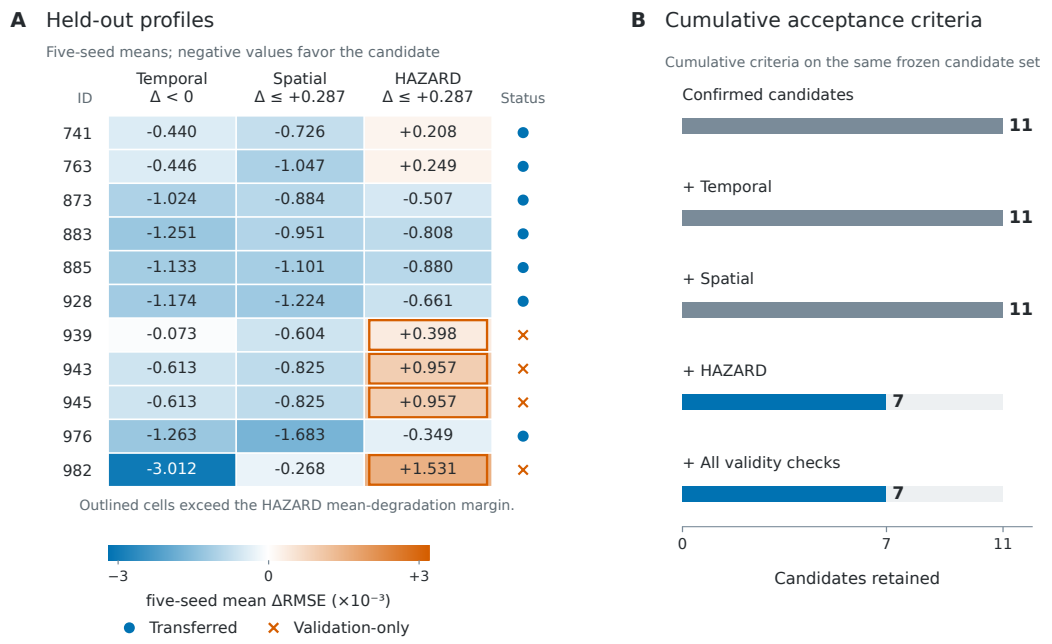


Figure 3: Held-out profiles and cumulative acceptance. Left: five-seed mean paired normalized-RMSE differences for all confirmed candidates, in units of 10^{-3}. Outlined cells exceed the HAZARD degradation margin. Right: cumulative application of the requirements retains seven candidates. HAZARD is a conditional slice of the temporal and spatial holdouts; IDs match Table 1.

seven after the HAZARD margin, and seven after all required final validity checks. The HAZARD condition excludes candidates 939, 943, 945, and 982; the spatial and validity conditions exclude no additional records from this confirmed set. These counts are computed from the same frozen profiles shown in Figure 3. They describe the contribution of each condition to the recorded decisions, without rerunning search or changing the historical verdicts.

D.2 Paired-Seed Intervals

Machine-learning comparisons can vary with data sampling, initialization, and hyperparameter choices (Bouthillier et al., 2021). The intervals below isolate training-seed variation for fixed candidate implementations and evaluation samples.

For a candidate and readout with paired differences $\{\delta_s\}_{s\in\mathcal{S}}$, define

$$\bar{\delta} = \frac{1}{5}\sum_{s\in\mathcal{S}} \delta_s, \qquad s_\delta^2 = \frac{1}{4}\sum_{s\in\mathcal{S}} (\delta_s - \bar{\delta})^2. \tag{13}$$

The descriptive two-sided 95% paired-seed interval is

$$\bar{\delta} \pm t_{0.975,4}\frac{s_\delta}{\sqrt{5}}, \qquad t_{0.975,4} = 2.776445. \tag{14}$$

These intervals characterize training-seed variation on the fixed sealed samples. They are pointwise summaries for individual candidate–readout pairs. Candidates share parent references and evaluation samples, so the intervals do not provide simultaneous coverage across the candidate set. The registered acceptance rule uses mean effects; interval endpoints do not determine the historical verdicts.

Table 17: Paired certification effects for all confirmed candidate records. Means, sample standard deviations, and interval endpoints are in normalized-RMSE units of 10^{-3}. Every row contains five valid paired seeds. The spatial and HAZARD acceptance margin is $+0.287$ in these units. Status refers to the complete joint rule.

ID	Readout	Mean Δ	SD	95% paired-seed interval	Status
741	Temporal	-0.440197	0.858369	$[-1.506002, 0.625609]$	Transferred
741	Spatial	-0.726343	0.364923	$[-1.179455, -0.273231]$	Transferred
741	HAZARD	$+0.207900$	1.774439	$[-1.995357, 2.411157]$	Transferred
763	Temporal	-0.445643	0.704726	$[-1.320676, 0.429390]$	Transferred
763	Spatial	-1.047438	0.304302	$[-1.425279, -0.669597]$	Transferred
763	HAZARD	$+0.249203$	0.737682	$[-0.666751, 1.165156]$	Transferred
873	Temporal	-1.023815	0.380016	$[-1.495667, -0.551962]$	Transferred
873	Spatial	-0.884101	0.131775	$[-1.047721, -0.720480]$	Transferred
873	HAZARD	-0.507123	0.586403	$[-1.235239, 0.220993]$	Transferred
883	Temporal	-1.250935	0.600485	$[-1.996536, -0.505334]$	Transferred
883	Spatial	-0.950637	0.201152	$[-1.200401, -0.700874]$	Transferred
883	HAZARD	-0.808354	0.929494	$[-1.962472, 0.345765]$	Transferred
885	Temporal	-1.133253	0.668644	$[-1.963484, -0.303022]$	Transferred
885	Spatial	-1.100606	0.155359	$[-1.293510, -0.907702]$	Transferred
885	HAZARD	-0.879808	0.872417	$[-1.963057, 0.203441]$	Transferred
928	Temporal	-1.173563	0.575430	$[-1.888053, -0.459072]$	Transferred
928	Spatial	-1.224251	0.112821	$[-1.364338, -1.084165]$	Transferred
928	HAZARD	-0.660538	0.925056	$[-1.809147, 0.488071]$	Transferred
939	Temporal	-0.073095	0.545903	$[-0.750923, 0.604733]$	Validation-only
939	Spatial	-0.604193	0.224410	$[-0.882835, -0.325550]$	Validation-only
939	HAZARD	$+0.397711$	1.125514	$[-0.999800, 1.795221]$	Validation-only
943	Temporal	-0.613221	0.182503	$[-0.839828, -0.386614]$	Validation-only
943	Spatial	-0.825360	0.291235	$[-1.186976, -0.463745]$	Validation-only
943	HAZARD	$+0.957093$	0.900215	$[-0.160672, 2.074857]$	Validation-only
945	Temporal	-0.612783	0.182984	$[-0.839988, -0.385578]$	Validation-only
945	Spatial	-0.825359	0.291128	$[-1.186843, -0.463875]$	Validation-only
945	HAZARD	$+0.956592$	0.901080	$[-0.162246, 2.075430]$	Validation-only
976	Temporal	-1.263088	0.349941	$[-1.697597, -0.828579]$	Transferred
976	Spatial	-1.683442	0.288993	$[-2.042275, -1.324610]$	Transferred
976	HAZARD	-0.348695	1.015400	$[-1.609481, 0.912091]$	Transferred
982	Temporal	-3.012184	0.793988	$[-3.998050, -2.026318]$	Validation-only
982	Spatial	-0.267678	0.415145	$[-0.783149, 0.247793]$	Validation-only
982	HAZARD	$+1.531017$	1.109790	$[0.153032, 2.909003]$	Validation-only

D.3 Mean-Based Acceptance and Estimate Precision

The complete profiles distinguish the acceptance decision from the precision of each estimated effect. Candidates 741 and 763 satisfy the registered mean-based rule, although their temporal intervals include zero and their HAZARD intervals extend above the permitted margin. Their favorable status therefore records compliance of the observed means with the joint rule.

Five candidates have negative means in all three held-out readouts: 873, 883, 885, 928, and 976. For each of these candidates, the HAZARD interval includes zero. The upper HAZARD endpoints for 873 and 885 remain below $+m$, while those for 883, 928, and 976 extend above it. The intervals thus provide additional resolution within the set of favorable mean profiles.

Candidate 982 has a temporal interval entirely below zero and a HAZARD interval entirely above zero. This contrast reinforces the difference between its temporal and conditional behavior. Its validation-only status follows its HAZARD mean exceeding $+m$; the interval is supplementary evidence rather than an additional acceptance test.

Candidates 943 and 945 have distinct source identities and slightly different full-precision certification effects. The agreement of candidates 943 and 945 at the displayed precision in Figure 3 does not imply identical full-precision results or independent mechanism discoveries.

D.4 Weather-Event Support and Scope of Uncertainty

The sealed samples are fixed across training seeds. Consequently, varying the seed measures how the trained models behave on those samples; it does not resample weather periods or event membership.

The custodian's support summary reports 3, 1, 1, and 12 HAZARD blocks for ZBAA, ZSPD, ZSSS, and ZBAD, respectively, at each horizon. As detailed in Appendix A.5, the blocks are constructed within airport–horizon pairs using the 90-minute linkage rule. They are not deduplicated across airports or horizons and are not assumed to be independent meteorological episodes.

This support matters because HAZARD aggregation gives equal weight to the airports despite their different event counts. The observations at ZSPD and ZSSS each arise from one recorded block. The reported means describe performance on this particular event-conditioned sample.

The aggregate support counts do not contain block-level paired errors and therefore cannot supply an event-resampling interval by themselves. No date- or event-block uncertainty interval is reported here. The paired-seed intervals and the event-support table describe complementary aspects of the available evidence without changing the original seven transferred and four validation-only verdicts.

E Objective Implementations and Development Effects

E.1 Implemented Objective Families

The seven transferred records span three broad design directions. Table 18 maps these records to their implemented modifications. The grouping summarizes shared design choices; it does not equate candidate counts with independent mechanism discoveries. The descriptions below retain the implementation differences within each direction.

Table 18: Design directions represented by the seven transferred candidate records. All listed records satisfy the joint held-out acceptance rule. The grouping describes implementations rather than isolated causal contributions.

Design direction	Candidate IDs	Implemented modification
Error-dependent reweighting	741, 763	Detached inverse batch-MSE weighting across horizons or horizon–component cells, with different normalization scales.
Horizon-conditioned robust penalties	873, 883, 885	Horizon-dependent residual penalties, with differences in coefficients and horizon treatment.
Horizon-conditioned robust penalties	928	Horizon-dependent pseudo-Huber terms combined with a quadratic penalty on out-of-range predictions.
Adaptive residual scaling	976	A pseudo-Huber penalty whose scale is computed from detached batch-residual statistics.

Error-dependent weighting. Candidate 741 computes detached inverse per-horizon batch-MSE weights and normalizes their mean to three. Candidate 763 extends the weighting to horizon–component cells and normalizes their mean to one. Both candidates improve the development aggregate and satisfy final acceptance. Their comparison changes weighting granularity and overall loss scale, so it does not isolate either factor individually.

Horizon-conditioned robust penalties. Candidate 873 retains squared error at the shortest horizon and uses robust penalties at longer horizons. Candidates 883 and 885 implement related horizon-conditioned variants. The implementations differ in their coefficients as well as their horizon treatment: the pseudo-Huber terms in 873 use a leading coefficient of two, whereas those in 883 and 885 use one. These distinctions are retained in the archived candidate sources.

Boundary-anchored robust loss. Candidate 928 combines horizon-dependent pseudo-Huber penalties with an out-of-range quadratic penalty on normalized predictions:

$$b(\hat{y}) = \mathrm{ReLU}(|\hat{y}| - 1)^2. \tag{15}$$

The resulting objective changes both residual curvature and the training penalty associated with predictions outside the prescribed range. The reported effect concerns this complete implementation.

Adaptive residual scaling. Candidate 976 replaces only the forecasting loss. The class-weight computation and classification loss are unchanged from the parent. Let prediction and target tensors have shape $B \times S \times H \times C$, with $H = 3$ forecast horizons and $C = 2$ wind components. Define $e_{bshc} = \hat{y}_{bshc} - y_{bshc}$ and let $M_{bs} \in \{0, 1\}$ indicate a valid runway slot. The mask is broadcast across horizons and components, and

$$N = \sum_{b=1}^{B} \sum_{s=1}^{S} M_{bs}.$$

For a nonempty valid batch, the parent uses masked MSE:

$$\mathcal{L}_{\text{parent}} = \frac{1}{HCN} \sum_{b,s,h,c} M_{bs} e_{bshc}^2. \tag{16}$$

The candidate computes a separate residual scale for each horizon–component cell. With $\varepsilon = 10^{-8}$, the frozen implementation calculates

$$q_{hc} = \frac{\sum_{b,s} M_{bs} e_{bshc}^2}{N + \varepsilon}, \tag{17}$$

$$r_{hc} = \mathrm{sg}(\sqrt{q_{hc}}), \tag{18}$$

$$d_{hc} = 2 \max(r_{hc}, 10^{-3}), \tag{19}$$

where sg denotes stop-gradient. Thus, six scales are recomputed for each batch by aggregating over its valid runway slots. The lower bound on d_{hc} is 2×10^{-3}; there is no upper clipping or moving-average state.

The forecasting objective is

$$\mathcal{L}_{976} = \frac{\sum_{b,s,h,c} M_{bs}\, 2d_{hc}^2 \left[\sqrt{1 + (e_{bshc}/d_{hc})^2} - 1\right]}{HCN + \varepsilon}. \tag{20}$$

The same validity mask is used for scale estimation and loss aggregation. All horizon–component cells enter the final average with equal explicit weight. The candidate adds no auxiliary regularizer and does not use the available `event_mask` or `event_available` arguments.

Because the scale is detached, the elementwise derivative used in backpropagation is

$$\frac{\partial \ell_{hc}(e)}{\partial e} = \frac{2e}{\sqrt{1 + (e/d_{hc})^2}}. \tag{21}$$

For $|e| \ll d_{hc}$, $\ell_{hc}(e) \approx e^2$ and its derivative approaches $2e$, matching the local MSE scale. For a fixed detached d_{hc}, the derivative magnitude approaches $2d_{hc}$ as $|e|$ increases. This is a batch-dependent bound: d_{hc} itself changes with the residual statistics, so the implementation does not impose a universal gradient bound across batches.

The reported results evaluate this complete replacement of masked MSE by an adaptive-scale robust objective. They do not separately identify the contribution of adaptive scaling relative to a fixed-scale robust loss. The frozen source identity is recorded in the candidate ledger (Appendix C).

E.2 Development Dispersion and Endpoint Coverage

All eleven confirmed candidates have five valid paired development results. Table 19 reports their mean effects, sample standard deviations, and endpoint coverage.

Table 19: Development effects and endpoint coverage for all confirmed candidates. Mean differences and standard deviations are in units of 10^{-3} normalized RMSE. Endpoint counts are out of eighteen.

Candidate	Mean Δ	SD	Negative mean	Negative in 5/5
741	-0.603	0.208	17/18	5/18
763	-1.110	0.128	18/18	10/18
873	-1.480	0.174	17/18	12/18
883	-1.547	0.249	17/18	12/18
885	-1.558	0.284	17/18	12/18
928	-1.674	0.271	18/18	12/18
939	-0.682	0.251	16/18	8/18
943	-1.033	0.374	15/18	10/18
945	-1.033	0.374	15/18	10/18
976	-1.857	0.278	18/18	16/18
982	-1.104	0.189	15/18	9/18

An endpoint is one airport–horizon–wind-component combination. A negative endpoint mean indicates that the five-seed average paired effect is favorable at that endpoint. The final column applies a stronger descriptive condition: the endpoint difference is negative under every seed. These endpoints share models and observations and are not independent experimental replications.

Candidates 763, 928, and 976 improve every endpoint mean. Candidate 976 additionally improves sixteen endpoints under all five seeds. This coverage shows how its aggregate improvement is distributed across the development task. Candidate 982 improves fifteen endpoint means but fails the sealed HAZARD margin, illustrating the distinction between development coverage and conditional transfer.

E.3 Horizon-Level Effects

Table 20 decomposes development effects by forecast horizon. Each entry averages paired differences over the three development airports, two wind components, and five seeds.

Table 20: Development horizon effects for all confirmed candidates. Values are paired normalized-RMSE differences in units of 10^{-3}.

Candidate	15 min	60 min	120 min
741	-0.315	-0.376	-1.117
763	-0.663	-1.144	-1.522
873	-0.513	-2.281	-1.647
883	-0.667	-2.144	-1.829
885	-0.589	-2.211	-1.874
928	-0.769	-2.304	-1.947
939	-0.238	-0.747	-1.062
943	-0.194	-1.363	-1.543
945	-0.194	-1.363	-1.542
976	-1.811	-1.841	-1.918
982	-0.855	-1.220	-1.238

Candidate 741 anticipates a predominantly short-horizon benefit, but its largest observed improvement occurs at 120 minutes. Candidate 873's proposed short-horizon emphasis likewise does not fully describe its measured profile. The longer-horizon expectations associated with 883 and 885 are consistent with their observed effects. Candidate 976 improves all three horizon means by similar amounts, although this aggregate pattern does not isolate its proposed effects on particular residual regimes.

The comparison between proposals and observations therefore supports two distinct assessments: whether the implementation improves performance, and whether the observed pattern agrees with its proposed explanation.

E.4 Sensitivity to the Screening Seed

Table 21 summarizes the confirmed candidates after excluding screening seed 42. All eleven retain negative differences under each of the four remaining seeds. Their four-seed means range from -0.000651 to -0.001779, and every mean exceeds the required improvement magnitude of 0.000574. Thus, the observed direction and mean magnitude of their gains extend beyond the screening observation.

This is a post-selection sensitivity analysis: the additional seeds also contributed to the original confirmation decisions. It assesses sign consistency and mean magnitude, without assigning a new four-seed statistical verdict or replacing the registered five-seed decisions.

Table 21: Development effects with and without screening seed 42. Paired normalized-RMSE differences are shown in units of 10^{-3}. All four-seed means satisfy the original mean-effect floor of -0.574, and all retained differences are negative. These descriptive summaries do not constitute new four-seed confirmation verdicts.

Candidate	Five-seed mean	Four-seed mean	Negative / four
741	-0.603	-0.651	4/4
763	-1.110	-1.131	4/4
873	-1.480	-1.479	4/4
883	-1.547	-1.454	4/4
885	-1.558	-1.453	4/4
928	-1.674	-1.565	4/4
939	-0.682	-0.730	4/4
943	-1.033	-0.996	4/4
945	-1.033	-0.995	4/4
976	-1.857	-1.779	4/4
982	-1.104	-1.183	4/4

E.5 Related Candidates and Feedback Chronology

Candidates 943 and 945 have distinct source digests, each fixed across its five development seeds. Their full-precision development scores differ, and their sealed profiles differ slightly despite coinciding at the precision displayed in Figure 3. They remain separate candidate records within a closely related range-barrier family.

The robust-objective sequence also illustrates the importance of feedback timing. Table 22 lists the relevant proposal and verdict times. Candidates 873, 883, and 885 were proposed after 871's screening result but before its confirmation refutation. Their available history could contain its promising screening observation, but not its later failure verdict.

Table 22: Selected events in the robust-objective sequence. All timestamps are UTC in September 2026.

Candidate	Event	Timestamp
871	Proposal	September 7, 07:02:50
873	Proposal	September 7, 10:34:38
883	Proposal	September 7, 18:56:46
885	Proposal	September 7, 23:59:54
871	Confirmation refutation	September 8, 12:12:37

F Supporting Controls and Lineage Analyses

F.1 Scope of the Supporting Studies

Three supporting analyses address different aspects of the discovery process. The finite-template reference evaluates static interventions under the common execution pipeline. The paired lineage

comparison examines executed proposal trajectories with and without prior development history. The context-ladder study measures automatically detected features of generated proposal text. Their outcomes are reported according to these distinct experimental units.

F.2 Finite-Template Coverage

Random search provides a standard reference for adaptive hyperparameter optimization (Bergstra & Bengio, 2012). Here, random sampling is applied to a finite inventory of executable templates, whose coverage and accounting are specified below.

The six-template registry contains 36 literal parameter combinations: five horizon-weight vectors, four event-weight settings, six class-focal settings, seven runway-component subsets, two cyclic encodings, and twelve multiscale-difference settings. The first three families concern downstream objectives; the remaining families concern representations.

The template inventory is broader than the 28 completed records in the frozen reference. Earlier regimes, calibration probes, failed executions, and reused configurations do not become additional members of that denominator. Appendix C.5 gives the exact membership and the disposition of the screening survivor 611.

The reference shares the parent, evaluator, screening seed, and downstream training budget with the LLM campaign. Its template space and generation process define the scope of the comparison. The same 28 records serve as one shared reference across the three proposal lineages.

F.3 Executed Lineage Trajectories

Reflexion uses retained feedback to guide subsequent agent trials (Shinn et al., 2023), while AgentRxiv enables research agents to share and build on previous results (Schmidgall & Moor, 2025). Our lineage comparison examines access to development history within a fixed model-editing setting.

The P01–P04 comparison schedules four rounds under each of two conditions, giving 32 proposal/run slots. Of these, 27 have valid primary numerical readouts, two have invalid primary readouts, and three have unavailable proposals or runs. Both conditions use the same Kimi proposal model, downstream-objective editing surface, parent reference, and seed-42 metric. The treatment concerns access to prior development lineage.

The invalid slots are P01-L-R03 and P02-L-R04. The unavailable slots are P02-L-R03, P04-L-R04, and P04-N-R03. Each ends in a proposal timeout before GPU training begins. They therefore contribute no numerical primary readout and zero training GPU-hours. All scheduled slots remain represented in the trajectory accounting; best-value summaries use valid readouts only.

Completed slots retain structured proposals, API-response records, and per-slot status metadata. The API logs record requested and served model identifiers, temperature settings, and thinking mode. The execution plan and final manifest specify the budget and stopping rules.

The inspected L and N prompt directories contain matching copies of the system instruction, common brief, code-generation template, and repair template. The experimental difference is the injected history. Recorded N slots have `history_chars=0` and a constant `prompt_sha256`. In the L condition, history is added from round 2 onward, and the recorded prompt digest changes across rounds.

History is stored as structured JSON and rendered at runtime by `history_renderer.py`. The inspected renderer digest matches the `renderer_sha256` recorded in the slot metadata. The `history_as_seen` directories are empty, so the fully rendered historical prompts were not preserved as text.

Per-slot digest fields are therefore available, but the exact rendered inputs cannot be directly inspected from saved prompt text. Moreover, the current common-brief file digest differs from the recorded `common_brief_sha256`; whether this reflects a file revision or a different hashing scope remains unresolved. The current template files are not treated as verified copies of every historical model input.

The observed trajectories differ across blocks. P01 includes repeated favorable macro-objective proposals in the no-lineage condition. P02 contains valid proposals in both conditions, with an unavailable third-round record and an invalid fourth-round readout in the lineage arm. All four no-lineage rounds have valid numerical readouts. In P03, the lineage condition refines a macro objective through successive scale changes, whereas the no-lineage condition explores several weighting mechanisms and also finds a favorable proposal.

Table 23 reports the best valid seed-42 difference in each arm of the four completed blocks. Unavailable or invalid rounds are omitted from the best-value selection and are not imputed.



I = invalid readout; U = proposal/run unavailable. Missing values are not imputed.

Table values use the same 10^{-3} units; bold values mark all tied best valid rounds.

L and N are offset horizontally within each round for visibility; vertical values are unchanged.

Figure 4: Proposal trajectories with (L) and without (N) development-lineage context for the completed P01–P04 blocks. Each arm has four scheduled rounds. Points are seed-42 paired differences for the candidate proposed in that round, in units of 10^{-3} normalized RMSE; they are not a running historical-best curve. I denotes an invalid readout, U an unavailable proposal or run, and a broken line a missing numerical value that is not imputed as zero. An outer ring marks the best valid round in that arm. The comparison describes mixed per-block patterns and does not establish a general advantage for lineage.

These records support a description of proposal evolution and observed development effects within the completed blocks. They do not provide a comparison of confirmation or transfer yield.

The recorded settings of this later comparison apply to that comparison alone. They are not used to fill missing sampling, retry, timeout, or model-serving fields in the main September 13 campaign.

Table 23: Best valid seed-42 paired differences in the completed P01–P04 lineage comparison. Values are in units of 10^{-3} normalized RMSE. Negative differences favor the candidate.

Block	Lineage best	No-lineage best
P01	$+0.010$	-0.796
P02	-0.478	-0.796
P03	-1.288	-1.087
P04	-1.235	-1.311

F.4 Context-Ladder Proposal Study

The context-ladder study contains 150 proposals: three language models, five context conditions, and ten generations per cell. The conditions provide task and interface information (L0), add requirements (L1), add measured evidence (L2), and then supply either raw lineage (L3) or curated lineage (L3c).

All 150 outputs parse successfully. The archived automatic scorer assigns one point for each of five features: mechanism-related wording, a numerical falsification cue, an evidence cue, endpoint specificity, and within-cell action-ID uniqueness. Parsing compliance is recorded separately.

Table 24 gives the mean score and sample standard deviation for each cell. An archived replay of the scorer reproduces the fifteen reported means. The score measures the presence of textual features; it does not verify the correctness of a mechanism or the accuracy of an evidence claim.

Table 24: Automatic proposal-feature scores out of five, reported as mean $\pm$ sample standard deviation over ten proposals per cell. These scores describe generated text rather than executed-model improvement.

Context	DeepSeek	GLM	Kimi
L0: task and interface	1.90 ± 0.32	1.80 ± 0.42	2.10 ± 0.57
L1: + requirements	2.90 ± 0.74	2.30 ± 0.48	2.40 ± 0.52
L2: + measured evidence	3.50 ± 0.85	3.20 ± 0.92	2.40 ± 0.84
L3: + raw lineage	2.70 ± 0.82	2.40 ± 0.52	2.90 ± 0.88
L3c: + curated lineage	2.70 ± 0.67	3.30 ± 0.67	2.20 ± 0.92

The patterns are model-dependent and non-monotonic. DeepSeek has its highest mean with measured evidence, GLM with curated lineage, and Kimi with raw lineage. The uniqueness feature depends on the other proposals in the same cell, so observations are not fully independent. No significance claim or trained-model performance gain is inferred from these scores.

G Reproduction Materials and Resource Accounting

G.1 Training Contract and Parent References

Parent architecture. The parent combines `UnifiedSeries2Vec` with `AeroWFForecastModel`. Scratch and pretrained forecasting legs construct the same architecture from the corresponding seed-specific pretraining configuration. Scratch initialization does not load the pretrained weights.

Table 25 summarizes the recorded architecture. The leading dimension in `Data_shape=[128,4,96,11]` is the batch size.

The forecasting model contains 3,932,907 parameters. Its optimizer includes 2,769,029 core parameters and 2,054 forecasting-head parameters, for 2,771,083 parameters in total. The remaining 1,161,824 parameters belong to other modules in the unified model and are not included in the forecasting optimizer.

The optimized core comprises the temporal encoder, frequency encoder, hierarchical module, exogenous encoder, and fusion module. No explicit `requires_grad=False` freezing is applied.

Table 25: Recorded v2p3 parent architecture. The forecasting head produces three horizons and two wind components for each runway.

Component	Configuration
Input	Four padded runways, 96 timesteps, 11 input channels
Temporal Transformer	Embedding size 128; representation size 256; three layers; four heads; feed-forward dimension 512
Frequency encoder	FreTS, two layers
Hierarchical module	Simple GNN enabled; hidden dimension 128; two layers
Exogenous encoder	METAR encoder enabled
Recorded dropout	0.2
Fusion	Residual addition; output dimension 256
Forecasting head	LayerNorm(256) followed by Linear(256, 6), applied per runway

Parameters excluded from the optimizer should therefore be distinguished from parameters explicitly marked as frozen. These parameter counts apply to the parent and candidates that preserve its architecture.

Downstream optimization. The four downstream legs use the common settings in Table 26. The scratch and pretrained legs differ in initialization and optimizer learning-rate groups. Although the saved scratch configuration also contains the CLI field `encoder_lr=3e-5`, the scratch optimizer uses `scratch_lr=1e-4`.

Table 26: Recorded downstream training settings. Encoder learning rate 3×10^{-5} applies only to the pretrained core.

Setting	Value
Optimizer	AdamW
Weight decay	10^{-4}
Scratch core/head learning rate	10^{-4} / 10^{-4}
Pretrained core/head learning rate	3×10^{-5} / 10^{-4}
Scheduler	CosineAnnealingLR; $T_{\max} = 60$, $\eta_{\min} = 0$
Scheduler update	End of each epoch
Downstream warmup	None
Epochs	60
Batch size	128
Gradient accumulation	None
Gradient clipping	3.0
DataLoader workers	0
`drop_last`	False
Training precision	FP32; AMP disabled
Hardware per trial	One RTX 3090; no DDP
Training early stopping	Disabled
Delivered checkpoint	`final_model.pth`
AdamW betas and epsilon	Not explicitly recorded

Forecasting training uses 56,574 complete-horizon samples. With batch size 128 and `drop_last=False`, each epoch contains 441 full batches and one batch of 126 samples, giving 442 optimizer steps. The 56,598-row training partition and event-flag files describe the source windows before this loss-specific filtering.

The optimizer calls do not explicitly supply betas or epsilon, and the archived configurations do not record their resolved values. These fields remain unspecified here rather than being reconstructed from current defaults.

Pretraining and initialization. The recorded pretraining configuration uses 100 epochs, learning rate 3×10^{-4}, and five warmup epochs. Its selected checkpoint is `best_model.pth`, distinct from the final-checkpoint policy for downstream training.

Parent runs and most candidate runs reuse existing seed-specific pretraining checkpoints. The corresponding pretraining runs were conducted on August 25–27 with a 100-epoch setting. The presence of a pretraining configuration does not imply a new pretraining run for every downstream candidate.

For scratch training, the model is constructed from the seed-specific configuration with PyTorch initialization and a randomly initialized forecasting head. For pretrained training, the corresponding seed-specific checkpoint is loaded with `strict=True`. The core and forecasting head are subsequently optimized.

Seeds and randomization. The seed set is $\{42, 43, 2027, 3407, 5519\}$. The training entry point seeds Python `random`, NumPy, PyTorch, and CUDA, and supplies a seeded generator for DataLoader shuffling. Dropout follows the PyTorch random-number state.

Deterministic algorithms are not explicitly enabled: the recorded implementation does not set `cudnn.deterministic` or call `use_deterministic_algorithms`. Seed control therefore should not be interpreted as a general guarantee of deterministic training.

Parent references. The applicable parent directories carry the `v2p3_enc3e5` identifier. Earlier `parent_local` directories and v2p3 directories without this qualification include different training regimes or learning-rate controls and are not substituted for these references.

Adjudication uses `parent_refs_v2p3/refs_seed{S}.json`. The five scratch-parent evaluator scores have mean 0.049355671 and sample standard deviation 0.000181382. For all 55 confirmed-candidate development results, the archived reference cross-check reproduces the recorded paired differences within 5×10^{-7}.

All confirmed candidates use the scratch comparison. Other scored candidates retain their recorded scratch or tier-2 decision field, as documented in Appendix C.

G.2 Historical Runtime and Code Versions

The D and OT1 parent records report Python 3.12.3, PyTorch 2.5.1+cu124, and PyTorch CUDA version 12.4. The recorded Linux kernels are 5.15.0-135 on D and 5.15.0-94 on OT1. The R-side pretraining environment records the same PyTorch wheel, while a complete downstream environment record is not available in the supplied extraction. Training uses FP32 without AMP.

The custodian reports that sealed inference used `device=cuda` on an NVIDIA GeForce RTX 4090, with CUDA runtime 12.4 and PyTorch 2.5.1+cu124. Inference used `float32`. These settings describe the executed certification environment, rather than the prediction script's default device.

The recorded repository commit is

`f7f549e5a1e1268d7016d7c5a851cc06d4d95b95`.

The archived common pipeline SHA-256 is

`74c93dcff102473be45fd90fd502d3d7b730515cd22e736dc21f83217eb384ae`,

and the forecasting training-entry SHA-256 is

`61077fae014baf8aee922563178be35f58f3f8556cc6c8ed82d73dd65bf8a5e9`.

These identify the common training implementation. Candidate-specific source identities are recorded separately in the scored-candidate index.

G.3 Historical Proposal-Model Settings

Available model-generation settings vary by campaign segment. Table 27 reports only historically recorded values. A configured model identifier does not establish complete per-request served-model coverage.

Table 27: Available main-campaign generation settings. A dash denotes an unavailable field. Segment labels describe archival coverage, not a claim that every request retained every field.

Recorded segment	Model identifier	Temperature	Token limit	Reasoning
D relay, 732 onward	`deepseek-v4-pro`	1.0	65,000	–
OT1 chain, record 820	`glm-5.3`	1.0	–	–
OT1 relay, record 840	–	–	30,000	–
OT1 relay, 870 onward	`glm-5.3`	0.75	65,000	high
R chain, record 910	`kimi-k3`	1.0	–	–
R relay, 942 onward	`kimi-k3`	1.0	65,000	–

Historical request timeouts, proposal and code-generation retry limits, smoke-repair limits, prompt digests, and the complete contemporaneous runner digest remain unavailable.

The separate September 22 lineage-control configuration records requested and served model `kimi-k3`, temperature 1.0 for both proposal and code generation, enabled thinking, and a token limit of 65,000. These settings describe that control experiment and are not used to fill missing main-campaign fields.

G.4 Artifact Identity and Evaluation Records

The candidate records connect proposals, implemented edits, parent references, training configurations, and observed outcomes. Archived edit digests identify the source associated with each scored candidate. Checkpoint-tensor identities and prediction digests provide additional checks on the trained models and submitted readouts.

The evaluator checks sample coverage and uniqueness, tensor shapes, finite values, scored-position ranges, and the required reproduction anchors. A serialized checkpoint-file digest checks file identity, while a checkpoint-tensor digest identifies the model state independently of serialization metadata. These checks bind reported effects to the artifacts being evaluated.

The September 13 freeze preserves the campaign lineages and confirmed reproduction bundles. The recorded frozen-ledger source SHA-256 is

`139ca875c227f066ff9742d2174e00ac4d6402a6184359fc3f80803fde46ea8f`

and the original normalized certification-mean readout SHA-256 is

`6a0023a30c4a769e9b98294c4c44f674aa1ff3269a770db942651bfc0288e984`.

These identify different archived objects: the campaign ledger and the subsequent certification readout. Later approved uncertainty and event-support summaries supplement the original verdicts.

G.5 Recorded and Unavailable Runner Fields

The original workflow separates proposal generation from code generation and includes rejection and repair handling. Historical settings are interpreted at the record or campaign-segment level. A current runner default is not evidence of the setting used for an earlier trial.

Table 28 summarizes the available provenance fields.

G.6 Runtime Fields and Accounting Scope

Archived trial metadata store `resource_usage.gpu_hours`, `resource_usage.gpu_model`, and `artifact_digests.edit_sha256`. The metadata audit found these fields in the result-record collections summarized in Table 29. Every recorded GPU model in this audit is `RTX3090`.

The field named `gpu_hours` records elapsed execution time on a single RTX 3090. It measures wall-clock GPU-hours rather than utilization-weighted compute.

Table 28: Availability of selected historical fields. Partial coverage is retained explicitly rather than completed using later runner settings.

Field	Available evidence
Scored-candidate edit digest	Available for all 70 records from frozen trial metadata.
Seed-42 decision value	Available where recorded and applicable; unavailable values remain missing.
Requested/served LLM identity	Partial coverage in main-campaign lineage; not a complete 70-record specification.
Sampling, retry, and timeout settings	Incomplete historical coverage; unavailable settings remain unspecified.
Prompt/context provenance	Main campaign: independent digests unavailable. Lineage control: per-slot prompt and renderer digests recorded, but rendered prompt text absent and common-brief hash scope unresolved.
Elapsed runtime and GPU model	Recorded in archived result metadata; coverage is reported separately below.
Later lineage-control settings	Available for parts of the separate comparison; not assigned retrospectively to the main campaign.

Table 29: Coverage of audited result metadata. These counts include records outside the 70 scored LLM candidates and are not campaign trial totals or runtime totals.

Archive collection	Result records
D	88
OT1	82
R	69
Clone	22

The reconciled records account for approximately 862 wall-clock GPU-hours across discovery, calibration, confirmation, and additional control studies (Table 30). This resource accounting is broader than the principal 70-record Regime B census. It includes earlier exploration and control executions that do not enter the reported candidate denominator.

The September 13 archive was deduplicated by trial identifier. For `730-c43`, this retained the confirmation rerun but omitted a distinct superseded execution. Both runs completed on September 8, 2026: the superseded run at 13:35:09 UTC consumed 3.350 hours, and the retained rerun at 20:42:42 UTC consumed 4.764 hours. The latter is already included in the frozen confirmation accounting; only the former is added separately. This adjustment recovers compute consumed before the freeze, rather than adding a post-freeze experiment.

Table 30: Recorded RTX 3090 wall-clock execution cost. The total is 861.850 hours, reported as approximately 862 GPU-hours in the main text. The scope includes earlier exploration and additional controls beyond the principal Regime B campaign. Executions without recorded durations and the exclusions described below are not included.

Accounting scope	GPU-hours
Regime A exploration and associated runs	87.951
Calibration probes 520–524	16.539
Regime B executions in the September 13 archive	458.014
Deduplicated frozen-archive subtotal	562.504
Post-freeze `rand-obj-611` confirmation (additional four seeds)	13.329
Separately recovered superseded `730-c43` execution (September 8)	3.350
P01–P04 lineage control	95.382
Additional failure-lineage ablation (v2)	174.329
Additional feedback ablation (v1)	12.956
Total recorded execution	861.850

Within the frozen Regime B accounting, LLM screening contributes 161.1 hours, confirmation contributes 249.9 hours, the original ten random-reference runs contribute 10.4 hours, and the clone execution collection contributes 36.6 hours. The clone collection includes the seed-42 screening run of `rand-obj-611`.

Recorded cost includes executions with invalid outputs and other charged attempts, so runtime-record counts need not equal scored-candidate or completed-trial counts. The P01–P04 lineage control schedules 32 slots. Twenty-seven produce valid primary readouts and account for 88.811 hours; two produce invalid primary readouts and account for 6.571 hours. These 29 GPU-executed slots total 95.382 hours. The remaining three slots terminate at proposal timeout, before training starts, and incur zero training GPU-hours. Invalid executions remain included in resource accounting.

The accounting excludes the five-seed parent pretraining, for which no `gpu_hours` field is available, proposal-only context-ladder generation, and custodian-side certification inference. Attempts without recorded durations, including the failed seed-43 replay for `rand-obj-611`, are also excluded. LLM API usage is not converted into GPU-hours. The total therefore describes recorded training-side execution rather than the complete end-to-end resource cost.

Recorded runtime for the supplementary random-arm confirmation. The screening run and four subsequent confirmation runs for `rand-obj-611` have 15.031 recorded RTX 3090 wall-clock hours in total. The four additional-seed runs account for 13.329 hours. Table 31 reports the individual records.

Table 31: Recorded runtime for `rand-obj-611`. Completion timestamps are UTC in 2026. Hours are elapsed wall-clock time on the RTX 3090 clone host, not utilization-weighted measurements. The subtotal excludes a failed seed-43 attempt whose runtime was not recorded.

Run	Completion time	Hours	Trial status
Seed 42 screening	Sep. 11, 03:36	1.702	Completed
Seed 43	Sep. 24, 16:43	3.333	Completed
Seed 2027	Sep. 24, 20:03	3.335	Completed
Seed 3407	Sep. 24, 23:23	3.328	Invalid; primary valid
Seed 5519	Sep. 25, 02:43	3.333	Completed
Additional-seed subtotal		13.329	
All five recorded runs		15.031	

An earlier seed-43 attempt, recorded on September 24 at 13:13 UTC, failed with a missing `metrics.json`. Its runtime is unavailable in the lineage. The reported subtotals therefore cover executions with recorded durations, rather than the complete cost of all attempts. The seed-3407 runtime is included because that execution consumed resources despite its invalid auxiliary output.

G.7 Reproduction Scope and Human Responsibilities

The evidence supports several distinct forms of verification. The frozen index and archived outputs support reconstruction of the reported accounting and promotion decisions. Executed source identities, training configurations, and seed-specific references support replay of fixed candidates. The sealed reproduction bundles support custodian-run evaluation within the protected data boundary.

Replaying the full adaptive proposal sequence is a different task. Its trajectory depends on language-model outputs, service versions, execution outcomes, and the development evidence available at each proposal. The retrospective artifact checks reported here do not constitute an independent rerun of the complete search campaign.

Human responsibilities include establishing the task and evaluation protocol, preparing interfaces, coordinating compute, documenting policy changes, and interpreting outcomes. The proposal agents generate the recorded hypotheses and executable edits. The custodian controls sealed evaluation and the release of approved summaries.

This division of responsibilities makes the scope of the framework explicit. The preserved records support inspection of how candidates were generated, evaluated, and accepted, while unavailable historical fields remain visible in the reproduction account.