Read, Grep, and Synthesize: Diagnosing Cross-Domain Seed Exposure for LLM Research Ideation

Yunju Choi [1] **Min Song** [2]

Abstract

The discovery of novel methodologies for emerging problems is a continuing cycle in ML, often driven by the migration of techniques across domains. We ask whether current LLM ideation systems benefit from *targeted* cross-domain retrieval or simply from exposure to diverse mechanisms. We study this question through PaperGym, a three-stage pipeline: (1) tool-augmented seed extraction from papers using read, grep, and bash, (2) cross-domain seed retrieval via paraphrasing across seven ML domains, and (3) method synthesis from retrieved seeds, each scored by rubric-based judges. Tool-augmented extraction improves specificity, and paraphrase-based retrieval broadens domain coverage. In synthesis, cross-domain retrieval receives more pairwise novelty wins than no-retrieval and same-domain baselines, but shows no significant difference from a random diverse-seed control, suggesting that current LLM ideation systems benefit from diverse seed exposure but do not yet reliably exploit the semantic rationale for retrieving particular seeds. We release the seed library, rubric prompts, and run scripts at https://github.com/yunjoochoi/PaperGym.

1. Introduction

Large language models are increasingly applied to research workflows (Huang et al., 2023): surveying literature, drafting code, running ML experiments, and proposing experimental directions. Among such tasks, the open-ended generation of novel research ideas remains a demanding task: the framings that yield strong ideas typically lie beyond direct retrieval, motivating reformulations that view the problem through alternative lenses.

[1]Yonsei University, Seoul, Republic of Korea [2]OnomaAI, Seoul, Republic of Korea. Correspondence to: Min Song <min.song@yonsei.ac.kr>.

AI4Research Workshop at ICML 2026. Copyright 2026 by the author(s).

One reliable source of novelty in human research is the migration of techniques across domain boundaries. Gradient-based adversarial optimization, prominent in adversarial examples for vision (Goodfellow et al., 2015), resurfaced as GCG (Zou et al., 2023) for LLM jailbreaking. Earlier work like SOLVENT (Chan et al., 2018) showed that human-annotated facet labels on research papers can support cross-domain analogy retrieval. Whether LLMs can systematically exploit such cross-domain migration during research ideation, beyond open-prompted analogy generation (Ding et al., 2023), has not been comparatively tested against simpler diverse-exposure alternatives.

We address this question with **PaperGym**, a three-stage pipeline: interactive seed extraction, cross-domain retrieval through paraphrasing, and method synthesis. A tool-augmented agent extracts grounded *seeds* from candidate papers via read, grep, and bash over an isolated sandbox. Seeds are short records of prior papers' problem settings and reusable method mechanisms; PaperGym retrieves them across seven ML domains and composes candidate methods. This compositional pattern parallels recent prompt synthesis (Liu et al., 2024; Xiong et al., 2025). We extend it to research ideation, with papers as the compositional primitives. Each stage is paired with a judge using rubrics for reusable scoring.

We focus on a single stage of the research workflow rather than full end-to-end research automation (Lu et al., 2024; Tang et al., 2025; Schmidgall et al., 2025), which runs over long horizons and complicates attribution of outcomes to specific design choices. Closer to our setting, ResearchAgent (Baek et al., 2024) targets ideation through a citation graph and an entity-centric knowledge store of shared concepts across papers, and SciMON (Wang et al., 2024b) retrieves inspirations from past papers and optimizes for novelty by iterating against literature. Unlike these systems, we explicitly paraphrase each problem into seven target-domain vocabularies before retrieval, and compose the retrieved seeds via attributed multi-seed synthesis.

To diagnose what drives ideation quality, we evaluate four ablation conditions on a 30-problem benchmark: no retrieval, same-domain retrieval, cross-domain retrieval, and a deliberately uninformative random-seed control. We inves-

tigate:

- **Extraction.** Does an agent equipped with read, grep, and bash over a paper sandbox extract more specific and well-grounded seeds than direct extraction without these tools?
- **Retrieval.** Does paraphrasing the problem statement across seven ML domains broaden the domain coverage of retrieved seeds without sacrificing relevance, compared to retrieval without paraphrasing?
- **Synthesis.** Does cross-domain seed exposure yield more novel methods than no-retrieval and same-domain baselines without sacrificing validity or coherence, and is any resulting gain driven by seed *content* or merely by the *presence* of a diverse seed pool?

Our findings are asymmetric: cross-domain retrieval receives more pairwise novelty wins than no retrieval (60% vs 40%) and same-domain retrieval (67% vs 30%, 3% tie), and matches a random-seed control (47% vs 53%). We take this as a diagnostic signal that current LLM ideation systems benefit from diverse mechanism exposure, but do not yet reliably exploit why particular seeds were retrieved. Section 4 returns to this distinction.

2. The PaperGym Pipeline

2.1. Tool-Augmented Seed Extraction

The first operation extracts *seeds* from candidate papers: short summaries of a method's underlying problem and its proposed mechanism. Each seed records a `(problem, method, domain)` triple anchored to a source paper.

Extraction runs as an LLM agent equipped with read, grep, and bash tools. The agent operates in an isolated sandbox containing the paper's full text in markdown form. If the paper provides an official code repository, the agent clones it dynamically for inspection during extraction. The agent first reads the paper end-to-end, identifies one to three distinct contributions, drafts candidate seeds, and then verifies each seed by locating its central claims (mechanism name, key design choice, or quantitative finding) via grep or additional reading; unverified claims are revised or dropped. This re-reading step distinguishes tool-augmented extraction from *direct extraction*, where the agent receives the entire paper as prompt context and emits seeds in a single pass without tool calls.

2.2. Cross-Domain Seed Retrieval

Given a research problem, the second operation retrieves seeds from the library across the seven ML domains (enumerated in Section 3.1). The original problem statement is paraphrased into six domain-specific restatements covering all ML domains except its natural domain. The natural domain is probed with the raw problem statement directly. Each restatement preserves the underlying problem while recasting it in that domain's native research vocabulary and concept structure (Figure 1 shows the Q23 case).

Each restatement (and the raw problem for the natural domain) is embedded and used to retrieve, by problem statement similarity, the top-k seeds from the library. Concretely, with embedding function ϕ, paraphrase $\rho_d(q)$ for domain d (setting $\rho_{d^\star}(q) = q$ on the natural domain $d^\star$), and the merged library $\mathcal{L}$, the seed pool retrieved for query q is

$$\mathcal{S}(q) \;=\; \bigcup_{d \in \mathcal{D}} \text{Top-}k\left(\phi(\rho_d(q)),\ \mathcal{L}\right), \tag{1}$$

with $|\mathcal{D}| = 7$, $k = 3$, and paper-id deduplication, so $|\mathcal{S}(q)| \leq 21$. We choose $k = 3$ in Eq. 1 to match the single-probe baseline's 21-seed budget. Retrieval is global across all domain shards, so the paraphrase vocabulary itself biases the top-k toward seeds from the target domain rather than partitioning by shard.

2.3. Multi-Seed Method Synthesis

The third operation composes the retrieved seeds into a candidate method for the original research problem. The synthesizer also produces a rationale articulating the methodological choices and an `inspired_by` list naming the seeds the method draws on, each annotated with the *borrowed aspect* taken from that seed. This structure encourages coherently composed, source-grounded synthesis. Each seed is presented with a *lens*: the text used to retrieve it, indicating why the seed was considered.

3. Empirical Study

3.1. Setup

Benchmark. We evaluate PaperGym on a benchmark of 30 research problem statements across seven ML domains: LLM/NLP, multimodal, computer vision, reinforcement learning, IR/recommendation, speech, and robotics. Each problem follows the form "*[gap]; how can we [research question]?*", combining a stated bottleneck with a directional research question.

Seed library. The retrieval library contains 1,167 seeds extracted by the tool-augmented agent from 446 conference papers (2017–2025) distributed across the seven domains. We provide the venue allowlist and per-domain budget in Appendix F. For a paired Stage 1 ablation (Section 3.2), we additionally extract seeds from 30 papers using direct prompting without tools. These direct-extraction seeds are scored only for Stage 1 quality and do not enter the retrieval

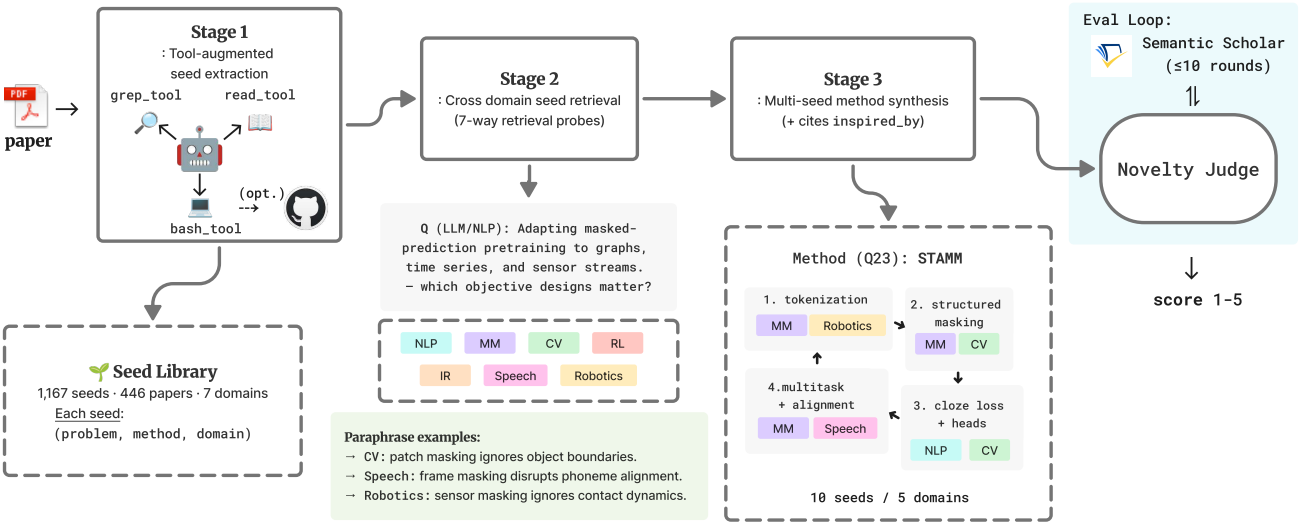

Figure 1. The PaperGym three-stage pipeline. *Stage 1* extracts grounded seeds from candidate papers using an LLM agent with read, grep, and bash over an isolated sandbox. *Stages 2–3* paraphrase the original problem across seven ML domains, retrieve top-k seeds per paraphrase from the resulting library, and synthesize a candidate method with attribution to the contributing seeds.

library.

Models. GPT-5 generates all outputs, and Claude Sonnet 4.6 serves as the judge. Different model families mitigate self-enhancement bias (Zheng et al., 2023), and pairwise positions are randomized per call to mitigate the position bias documented for LLM judges (Wang et al., 2024a).

Rubrics. Each stage is scored by rubric-based rating judges (Liu et al., 2023); Stage 3 additionally uses pairwise judges with positions randomized per call. Stage 1 rates seed-level *specificity* and *grounding* (1–5); Stage 2 reports *domain coverage* (1–7) and *relevance* (naive and lens-aware, 1–5); Stage 3 rates method-level *novelty* and *validity* (1–5), plus pairwise novelty/validity/coherence and auxiliary *seed incorporation* (1–3) and *attributed domain coverage*. Full rubric definitions appear in Appendix D.

Ablation. For Stage 3 we compare four ablations: **A** (no retrieval, problem only); **B** (same-domain retrieval, top 21 seeds from the problem's domain); **C** (cross-domain retrieval, top-k seeds aggregated across the seven paraphrase domains); and **D** (random-seed control, 21 seeds drawn uniformly from the library). A, B, and C progressively broaden the retrieval scope while keeping the seed budget fixed at 21 from B onward. D tests whether the gains observed under C depend on seed content or merely on seed presence.

3.2. Stage 1: Tool Augmentation Improves Seed Specificity

We compare *tool-augmented extraction* (Section 2.1) against a *direct extraction* baseline that emits seeds in a single prompt without tool calls, on 30 paired papers. Each paper's seed pool is scored along the Stage 1 rubric and aggregated to a paper-level mean.

Specificity. Tool augmentation lifts mean seed specificity from 4.22 ± 0.57 (direct extraction) to 4.76 ± 0.22 (tool-augmented extraction), a $+0.54$ absolute gain with markedly lower spread. This is consistent with interactive paper-environment exploration (navigating code, and re-reading sections of the paper to verify each seed's grounding, much as a human would) producing more concretely specified seeds than direct prompting alone.

Grounding. Both conditions score similarly on grounding (4.82 in both settings), indicating that the gain in specificity does not come at the cost of factual fidelity to the source paper.

Negative control. To verify that the grounding rubric responds to actual fidelity rather than reflecting judge leniency, we re-ran the grounding judge on a shuffled-paper control where each seed is paired with a random unrelated paper. Grounding collapses to 1.00 in 100% of cases (n=85 for direct extraction, n=82 for tool-augmented extraction; mean drop ≈ 3.82). High grounding scores therefore reflect real paper-seed alignment, not indiscriminate high marks.

3.3. Stage 2: Paraphrasing Achieves Complete Cross-Domain Coverage

We compare two retrieval modes that share a 21-seed budget: paraphrase mode (top-3 per domain) and a single-probe baseline (top-21 from the original problem).

Coverage. Paraphrase mode achieves domain coverage of 7.00 ± 0.00: every one of the 30 problems retrieves seeds spanning all seven domains. Single-probe retrieval reaches 5.03 ± 1.22. Paraphrasing improves coverage in $28/30$ problems; the two non-improvements already span all seven domains without paraphrasing.

Relevance. On the 1–5 relevance rubric (Appendix D), paraphrase mode scores naive 1.73 and lens-aware 1.91; the single-probe baseline scores 2.16 (lens = naive by construction). The lens-aware lift shows the paraphrase frame helps the judge recognize cross-domain analogical fit that surface vocabulary alone obscures. Paraphrase mode still trades per-seed relevance for full coverage relative to single-probe. We next ask whether this broader substrate translates into better synthesis.

3.4. Stage 3: Diverse Exposure Helps, Targeting Remains Unresolved

We measure the quality of synthesized methods across conditions A–D (defined in Section 3.1). Pairwise novelty is the primary axis; rating-scale scores, validity, and attributed domain coverage serve as supporting evidence. The novelty judge follows a ReAct loop (Yao et al., 2023) that augments its rating with Semantic Scholar (Kinney et al., 2023) searches (up to 10 rounds), letting it check the synthesized method against published prior work rather than rely on parametric memory alone.

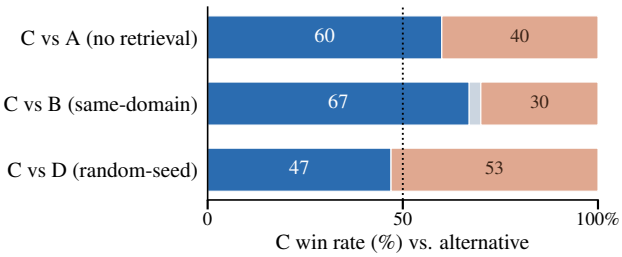

Figure 2. Stage 3 pairwise novelty win rates ($n = 30$; positions randomized). C trends above A and B but is on par with D.

Pairwise novelty. Cross-domain seed exposure (C) receives more novelty wins than both no-retrieval (A) and same-domain retrieval (B); relative to the random diverse-seed control (D), however, C neither wins nor loses at this sample size (Figure 2). Per-condition novelty averages mirror this picture (A: 3.97, B: 4.07, C and D tied at 4.13).

Validity. Per-condition validity averages cluster near ceiling (4.97–5.00 across all conditions) and provide little discriminative signal. The pairwise validity breakdown appears in Appendix E.

Coherence. We additionally ran a pairwise coherence judge measuring whether each method's mechanism is internally consistent and implementable as written. C wins $17/30$ against D (binomial $p \approx 0.58$). Random-seed methods are thus not detectably less coherent than cross-domain ones; per-condition single-pass coherence shows the same C-vs-D parity (Table 5).

Attribution integration. Both C and D surface seeds from multiple domains, but cross-domain attributions are more reliably integrated: only $1/30$ problems under C contain a seed attributed yet not incorporated, against $8/30$ under D (paired McNemar's exact small-sample test Eq. 4, with χ^2 form Eq. 3; $p \approx 0.016$). Per-condition means and the discordant-pair breakdown appear in Appendix E.

4. Discussion

Our results support a diagnostic conclusion rather than a simple claim that cross-domain retrieval works: condition C receives more novelty wins than both no retrieval and same-domain retrieval, consistent with broader seed exposure helping synthesis, but these gains disappear against the random diverse-seed control. The parity with D points to a concrete catalyst regime for current LLM ideation systems: they appear to benefit from being handed a diverse set of mechanisms, whether targeted or random, yet do not reliably convert the semantic reason those seeds were retrieved into higher judged novelty or coherence. The single C-over-D effect we detect is attribution integration ($1/30$ vs $8/30$ problems with abandoned citations, $p \approx 0.016$): evidence of some sensitivity to retrieval rationale, but not enough to produce a clear downstream advantage on our main judged outcomes. PaperGym therefore contributes a reproducible evaluation protocol for diagnosing when cross-domain seed exposure helps, and where targeted retrieval has yet to deliver.

5. Limitations and Future Work

The benchmark covers 30 problems judged by one model family (Sonnet 4.6); repeated random-pool resampling, broader benchmarks, multi-judge or human-expert evaluation (Si et al., 2024), and a larger seed library would sharpen the conclusions and expose a wider range of cross-domain mechanisms. PaperGym is designed to support a rubric-rewarded training environment for Stage 3, using the random-seed control as a non-trivial baseline under the observed C-vs-D parity. Other directions include self-extending seed libraries, retrieval and synthesis policies that make targeted seed content matter more than diverse exposure alone, and researcher-in-the-loop ideation workflows.

Software and Data

We release the PaperGym seed library, rubric prompts, and run scripts at https://github.com/yunjoochoi/PaperGym. The release includes the fixed random seeds and configuration needed to reproduce the reported retrieval and synthesis conditions.

Acknowledgements

We are grateful to the reviewers for their careful reading and constructive suggestions.

Impact Statement

This work studies LLM-driven research ideation, which can accelerate research output but introduces concerns about idea attribution. PaperGym addresses this by recording per-seed provenance in every synthesized method, making the source of each borrowed mechanism inspectable. Broader risks (displacing human ideation, concentrating research direction, inheriting biases from the seed library) warrant attention in future deployments.

References

Baek, J., Jauhar, S. K., Cucerzan, S., and Hwang, S. J. ResearchAgent: Iterative research idea generation over scientific literature with large language models, 2024.

Chan, J., Chang, J. C., Hope, T., Shahaf, D., and Kittur, A. SOLVENT: A mixed initiative system for finding analogies between research papers. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW), 2018. doi: 10.1145/3274300.

Ding, Z., Srinivasan, A., MacNeil, S., and Chan, J. Fluid transformers and creative analogies: Exploring large language models' capacity for augmenting cross-domain analogical creativity. In *Conference on Creativity and Cognition (C&C)*, 2023. doi: 10.1145/3591196.3593516.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

Huang, Q., Vora, J., Liang, P., and Leskovec, J. MLAgentBench: Evaluating language agents on machine learning experimentation, 2023.

Kinney, R., Anastasiades, C., Authur, R., Beltagy, I., Bragg, J., Buraczynski, A., Cachola, I., Candra, S., Chandrasekhar, Y., Cohan, A., et al. The semantic scholar open data platform, 2023.

Liu, X., Xu, N., Chen, M., and Xiao, C. AutoDAN: Generating stealthy jailbreak prompts on aligned large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., and Zhu, C. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. doi: 10.18653/v1/2023.emnlp-main.153.

Lu, C., Lu, C., Lange, R. T., Foerster, J., Clune, J., and Ha, D. The AI scientist: Towards fully automated open-ended scientific discovery, 2024.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B. P., Hermann, K., Welleck, S., Yazdanbakhsh, A., and Clark, P. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Schmidgall, S., Su, Y., Wang, Z., Sun, X., Wu, J., Yu, X., Liu, J., Moor, M., Liu, Z., and Barsoum, E. Agent laboratory: Using LLM agents as research assistants. In *Findings of the Association for Computational Linguistics: EMNLP*, 2025.

Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Si, C., Yang, D., and Hashimoto, T. Can LLMs generate novel research ideas? a large-scale human study with 100+ NLP researchers, 2024.

Tang, J., Xia, L., Li, Z., and Huang, C. AI-researcher: Autonomous scientific innovation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., and Sui, Z. Large language models are not fair evaluators. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024a.

Wang, Q., Downey, D., Ji, H., and Hope, T. SciMON: Scientific inspiration machines optimized for novelty. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024b.

Xiong, C., Chen, P.-Y., and Ho, T.-Y. CoP: Agentic red-teaming for large language models using composition of principles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., and Fredrikson, M. Universal and transferable adversarial attacks on aligned language models, 2023.

A. Q23 Pipeline Walkthrough: STAMM

This appendix traces a single condition C run of the standard pipeline (Section 2) on Q23 (cross-modal masked pretraining): paraphrase the problem across six non-natural ML domains, retrieve 21 seeds via seven probes (top-3 each) against the merged library, and synthesize a candidate method with cross-domain attributions. We compare the synthesized method against the no-retrieval baseline (condition A) below; the C-vs-D comparison on Q23 appears in Appendix B, and the loop-based ablation across all 30 problems appears in Appendix C.

Problem statement.

> *"Masked-prediction pretraining transformed language and vision; how can we adapt the same principle to other data types like graphs, time series, or sensor streams, and what objective design choices matter?"*

Pipeline summary. A single condition C synthesis on Q23 (paraphrase → retrieve 21 seeds via seven probes → synthesize) produced *STAMM* in approximately 74 s of GPT-5 wall-clock at \$0.07 of API cost (paraphraser 29 s, synthesizer 45 s; standard pricing). The novelty judge (Section 3.4) subsequently rated STAMM at $4/5$ on novelty. We use the walkthrough below as an illustrative trace of how the C-condition pipeline composes ten attributed seeds into a concrete method.

Synthesized method. *STAMM: Structure- and Time-Aware Masked Modeling* is a four-stage pretraining recipe that adapts masked prediction to graphs, time series, and multi-rate sensor streams:

Stage 1: Modality-specific tokenization. Lightweight adapters turn each modality into sequences with explicit structural or temporal indices. A graph adapter encodes nodes and edges as tokens with type and positional codes over the graph topology. A time-series adapter emits event tokens with timestamps and optional segment IDs. A fusion adapter for asynchronous sensors initializes a continuous-time latent via a Neural CDE and samples tokens at observed times, so a single shared transformer can attend over irregular, multi-sensor events. This Stage 1 CDE latent serves as the continuous-time context over which Stage 3's cloze loss is computed for irregular sensor streams.

Stage 2: Structure- and causality-aware masking. For graphs, mask nodes, edges, and small motifs as whole segments with synchronized edge-node masking. For time series, combine span masking with a prefix-style objective: bidirectional within a prefix window, forecasted thereafter. For multi-rate sensors, interleave synchronized masks (drop aligned timestamps across channels) and channel-specific spans. Mask spans are sampled at multiple granularities and snapped to detected boundaries to avoid pathological holes.

Stage 3: Energy-based cloze with decoupled heads. Adopt an energy-based cloze loss trained with conditional noise-contrastive estimation: a learned, modality-aware noise process proposes replacements for masked tokens, and the model discriminates real from noised values, links, or attributes. Decoupled high-capacity token heads (MLP for continuous values; classification for discrete attributes/links) prevent the backbone from overfitting low-level details. Auxiliary objectives align with each structure: link prediction and motif classification for graphs; temporal order recovery and derivative matching for time series; cross-channel contrastive agreement for sensors.

Stage 4: Alignment, curriculum, and curation priors. A monotonic-alignment regularizer encourages attention to progress roughly with time for streaming data. Cross-stream attention links let one decoder pathway attend to another stream's token-level states, tightening attribution across heterogeneous sensors. STAMM is trained in a multitask, instruction-style schema mixing imputation, forecasting, link prediction, and segmentation. Dataset sampling follows compute-aware filtering: each pool gets a utility and repetition half-life, and steps are allocated to maximize expected utility under a fixed budget; mask rates and span lengths anneal short-to-long.

Attributed seeds. STAMM cites ten seed attributions across five domains (Table 1). Multimodal contributes the shared-backbone scaffolding and curation/instruction priors; CV contributes the prefix-style hybrid attention and decoupled prediction heads; LLM/NLP contributes the energy-based cloze objective; Robotics contributes the Neural CDE for asynchronous sensors; Speech contributes the monotonic alignment and cross-stream attention.

Cross-domain composition. The walkthrough's most distinctive cross-domain bridge is the Neural CDE, attributed in our library to a Robotics seed (Multiscale Sensor Fusion with Neural CDEs), used as the continuous-time tokenization over which Stage 3's energy-based cloze loss operates for irregular sensor streams. Each individual borrowed mechanism

Table 1. Seeds attributed in STAMM under condition C, with `borrowed_aspect` mapped to the corresponding STAMM component. Two source papers (ViLT, AIM) each contribute two distinct seeds; their attributions are merged into a single row, so the table's eight rows cover ten attributions.

Domain	Source paper	How STAMM uses the seed
Multimodal	ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision (`arXiv:2102.03334`)	STAMM uses shallow modality adapters as the basis for Stage 1 tokenization and adapts whole-segment masking to graph segment masks in Stage 2.
Multimodal	OFA: Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework (`arXiv:2202.03052`)	Its multitask instruction-style pretraining schema informs Stage 4's mixed-task training setup.
Multimodal	Scaling Laws for Data Filtering – Data Curation cannot be Compute Agnostic (`arXiv:2404.07177`)	STAMM uses compute-aware filtering, utility scores, and repetition half-life as curation priors in Stage 4.
CV	Scalable Pre-training of Large Autoregressive Image Models (AIM; `arXiv:2401.08541`)	Its PrefixLM-style attention motivates the Stage 2 prefix objective, and its decoupled token-level prediction head motivates the Stage 3 heads.
LLM/NLP	Pre-Training Transformers as Energy-Based Cloze Models (`arXiv:2012.08561`)	STAMM adapts the energy-based cloze objective with conditional NCE as its Stage 3 prediction loss.
Robotics	Multiscale Sensor Fusion and Continuous Control with Neural CDEs (`arXiv:2203.08715`)	STAMM uses a Neural CDE latent to build the Stage 1 adapter for asynchronous multi-rate sensor streams.
Speech	Understanding the Modality Gap: An Empirical Study on the Speech-Text Alignment Mechanism of Large Speech Language Models (`arXiv:2510.12116`)	Its token-alignment analysis motivates the monotonic alignment regularizer used for time-indexed sequences in Stage 4.
Speech	DNCASR: End-to-End Training for Speaker-Attributed ASR (`arXiv:2506.01916`)	STAMM adapts cross-attention linkage as Stage 4 cross-stream connections between token-level states.

has a published precedent, but the rubric's Semantic Scholar search did not surface an end-to-end clone of this particular combination. We note that continuous-time latents (latent ODE/CDE families) have an active line of work for irregular time-series imputation that the rubric's search did not surface, so the novelty claim here is relative to the prior work that the rubric surfaced rather than a global one.

Comparison without retrieval (condition A). Running Q23 through condition A (no retrieval, problem statement only) yields *GMRP* (Generalized Masked Reconstruction Pretraining), an in-domain method combining Graph Transformers with Laplacian and random-walk encodings, a Temporal Transformer/TCN hybrid for time series, and a cross-sensor transformer for asynchronous events; multi-granular masking (node attribute / edge / subgraph; span / spectral patch; channel-drop) feeds a modality-aware decoder with mixture-density and spectral losses. All cited mechanisms are standard, well-defined components from existing masked-pretraining literature; no cross-domain transfer occurs.

Pairwise assessments (STAMM vs A). Under the same pairwise rubrics as Section 3.4 (Sonnet 4.6 judge with position randomization), the assessments are mixed (Table 2): STAMM is preferred for novelty, GMRP is preferred for validity and coherence.

The novelty preference reflects STAMM's cross-domain bridges: mechanisms attributed to Robotics (Neural CDE) and Speech (monotonic alignment, cross-stream linkage) seeds in our library are foreign to the prior work the rubric's search surfaced. GMRP, by contrast, stays within in-domain objective-design space, which the judge favored on validity and coherence; STAMM trades that within-domain tightness for the cross-domain combination noted above. The balance varies problem by problem in the aggregate benchmark, and this single trace should not be read as the universal pattern.

Table 2. Q23 pairwise assessments: STAMM (condition C) versus the no-retrieval baseline GMRP (condition A), under the same rubrics used in the main benchmark.

Axis	Winner	Judge's pivot reason
Novelty	STAMM (C)	"Energy-based cloze loss trained with conditional noise-contrastive estimation for continuous masked targets, combined with the Neural CDE-based fusion adapter for asynchronous multi-rate sensors and the prefix-style hybrid bidirectional/causal attention objective."
Validity	GMRP (A)	The judge cites GMRP's "more systematic and comprehensive ablation framework explicitly mapping the objective design landscape" of in-domain choices; STAMM's narrower focus on the cross-domain combination is judged "less directly tied to the modality-specific distributional challenges."
Coherence	GMRP (A)	The judge flags STAMM's "integration of Neural CDE outputs into the tokenization pipeline is underspecified" and that its energy-based cloze loss requires "a separate noise/proposal model whose design and training procedure are never specified."

B. Q23 C-vs-D Comparison

The D-condition synthesis on Q23 (random-seed control, 21 seeds drawn uniformly from the library) produced *MaskSCM: Masked Pretraining for Structured and Continuous Modalities*, a four-component recipe: (1) modality-aware tokenization yielding semantically aligned tokens (nodes/edges/motifs for graphs; variable-length segments or multi-scale patches for sequences; coarse "event" tokens for asynchronous sensors); (2) structure- and span-aware masking curricula; (3) prediction heads and losses matched to each modality's physics/statistics, including continuous-latent reconstruction when discretization is lossy; (4) a hierarchical backbone that propagates positional/structural encodings through depth, with cross-modal guidance when multiple sensors are available. The synthesizer further specifies iterative masked decoding, prototype-dissimilarity and uncertainty regularizers, and modality-specific backbone choices (state space blocks, Transformers, or graph Transformers). MaskSCM also receives novelty score $4/5$, with the synthesizer producing it in approximately 67 s of GPT-5 wall-clock at $0.06 of API cost (D condition skips the paraphraser).

Pairwise assessments. Under the same pairwise rubrics as Section 3.4 (Sonnet 4.6 judge with position randomization), STAMM (C synthesis from Appendix A) and MaskSCM (D) split the three axes (Table 3): STAMM wins novelty, MaskSCM wins validity and coherence.

Table 3. Q23 pairwise assessments. STAMM (C) wins novelty; MaskSCM (D) wins validity and coherence under the same rubrics used in the main benchmark.

Axis	Winner	Judge's pivot reason
Novelty	STAMM (C)	STAMM's "energy-based NCE cloze loss applied to masked prediction across heterogeneous modalities is harder to find in prior literature as a unified design choice"; the Neural CDE fusion adapter combined with prefix-style hybrid masking is also "a more distinctive mechanistic combination."
Validity	MaskSCM (D)	The judge cites MaskSCM's "explicit matching of objectives to each modality's physics/statistics" of in-domain choices; STAMM's energy-based NCE is judged "justified more on efficiency grounds than on directly resolving the core challenge."
Coherence	MaskSCM (D)	The judge flags STAMM's energy-based NCE as requiring "a well-defined noise distribution that is non-trivial to specify and train stably," noting that "the method does not explain how this noise process is initialized or learned without collapse." MaskSCM's coherence gaps are flagged as "more about underspecification of regularizers."

As in Appendix A, this is a single per-problem trace; it illustrates the main-benchmark C-vs-D pattern (Section 3.4) at this scale: targeted cross-domain retrieval can secure a novelty win without simultaneously delivering on validity and coherence. The 30-problem aggregate remains the reference point for generalization.

C. Novelty Iteration Behavior

This appendix wraps the standard pipeline in a novelty-guided iteration loop, in the spirit of self-feedback frameworks such as Self-Refine (Madaan et al., 2023) and Reflexion (Shinn et al., 2023), and reports an aggregate across all 30 problems. Each round, the synthesizer's output is scored by the novelty judge (Section 3.4); if the score is below threshold (4 in this

benchmark), the judge's reasoning and surfaced prior work are passed back to the synthesizer for re-synthesis (up to 10 rounds).

Aggregate. The iteration loop is rarely triggered: $28/30$ C-condition problems and $26/30$ D-condition problems clear the novelty threshold (4) on the first synthesis attempt (mean rounds C: 1.07, D: 1.13), so the loop exits without any feedback round in the vast majority of problems. This reflects a property of the evaluation rather than of the synthesizer: the single-pass novelty distribution is concentrated at score 4 (over 24 of 30 problems in each of conditions C and D), so a ≥ 4 threshold rarely admits drafts into the refinement loop. The few problems requiring a second round reach near-perfect scores (C: 5.0 on n=2, D: 4.75 on n=4), and both conditions arrive at mean novelty score $4.1/5$. Pairwise comparisons across all 30 problems under the same rubrics as Section 3.4 show no separation between C and D on any axis (Table 4).

Table 4. Loop ablation pairwise win rates over 30 problems. Binomial p computed via Eq. 2. None of the three axes shows a significant C-vs-D difference; under our novelty rubric the loop is rarely triggered (see *Aggregate* paragraph above).

Axis	C wins	D wins	Ties	Binomial p
Novelty	50%	50%	0%	1.00
Validity	40%	30%	30%	0.66
Coherence	47%	53%	0%	0.86

D. Rubric Definitions

Stage 1. Seed-level rubrics on a 1–5 scale. *Specificity* measures the density of concrete technical entities in the seed's method text, drawn from six categories: named algorithms, datasets/benchmarks, hyperparameter + value, architecture components, equations/symbols, and procedural steps (verb + object). Anchors scale by count: 1 for 0–1 entities, 5 for eight or more drawn from at least two distinct categories that include a named algorithm, a hyperparameter value, or an equation; procedural-verb-only does not reach 5. *Grounding* measures whether each entity in the seed has an equivalent in the source paper (paraphrase allowed). Hallucinations cap at 1, invented wording at 2–3 by count, and 5 requires verbatim/paraphrase match plus claim fidelity (no overstating, hedges preserved).

Stage 2. The primary measure is *domain coverage*: the number of distinct ML domains (1 to 7) among the retrieved seeds. Two auxiliary 1–5 rubrics score individual seeds. *Naive relevance* sees only the research problem and the seed's `(problem, method)` text (domain label withheld) and scores mechanism transferability: 5 applies as-is or addresses the same gap from a different angle, 4 reusable with non-trivial adaptation, 3 thematic overlap without an actionable lever, 2 shared area only, 1 unrelated. *Lens-aware relevance* additionally sees the lens text (the raw problem statement for single-probe and same-domain slots, the domain-specific paraphrase for cross-domain slots) and is explicitly directed to ignore lens eloquence: a fluently written lens cannot rescue a seed whose mechanism is genuinely unrelated. For single-probe retrieval the lens equals the raw problem statement, so the two scores coincide by construction; for paraphrase mode they can diverge, and that divergence isolates what the paraphrase contributed beyond surface restatement.

Stage 3. *Novelty* (1–5) is judged via a ReAct loop (Yao et al., 2023) with up to 10 Semantic Scholar queries: 1 identical to a published technique on the same problem, 2 surface variant of a parent technique, 3 known mechanism combination with prior joint application, 4 combination not surfaced as joint OR a named new component, 5 a new mechanism plus no direct near-clone. The retrieved seeds are static context visible to the judge, not a separate scoring axis, so the random-seed control (D) does not receive a structural penalty. *Validity* (1–5) asks whether the method's mechanism plausibly engages the research problem's failure mode; feasibility, soundness, novelty, and empirical effectiveness are excluded. *Pairwise judges* on novelty, validity, and coherence compare two methods with positions randomized per call (seed 42); each axis focuses on a distinct dimension (strongest novel element / most direct failure-mode engagement / most serious flaw), and all prompts ignore verbosity and break ties only when the methods are genuinely indistinguishable. *Inspired-by incorporation* (1–3: no/partial/full) checks whether each cited seed's `borrowed_aspect` is integral to the method, with the hard test "would removing it change the mechanism?". We additionally compute *attributed domain coverage* as a structural confirmatory signal: the number of distinct domains among the seeds the synthesizer explicitly attributes in its `inspired_by` field. All point rubrics default to the lower score on adjacent-anchor ties, suppressing optimism bias.

E. Single-Pass and Pairwise Results

This appendix reports the full per-condition averages and C-centered pairwise assessments that support the main benchmark (Section 3.4), plus supplementary numbers not included in the main text.

Statistical tests. Pairwise win counts are summarized via two-sided binomial tests on decisive outcomes (ties excluded). With n decisive comparisons and w wins for the condition of interest,

$$p_{\text{two-sided}} = \min\left(1,\ 2 \cdot \min\left(\sum_{i=0}^{w} \binom{n}{i} 2^{-n},\ \sum_{i=w}^{n} \binom{n}{i} 2^{-n}\right)\right). \tag{2}$$

For paired contrasts on inspired-by attributed-domain coverage between conditions C and D, we use McNemar's test on the discordant pair counts b (problems incorporated only under C) and c (only under D). The standard χ^2 statistic is

$$\chi^2 = \frac{(b-c)^2}{b+c}, \tag{3}$$

but for small $b + c$ we report the exact two-sided binomial McNemar

$$p_{\text{exact}} = 2 \cdot \sum_{i=0}^{\min(b,c)} \binom{b+c}{i} 2^{-(b+c)}, \tag{4}$$

which yields the $p \approx 0.016$ cited in Section 3.4.

Single-pass per-condition averages. For each condition we sample one method per problem (n=30) and score it on the held-out novelty and validity rubrics (Section 3.4); we additionally apply the method-level specificity rubric on a 1–5 scale (the Stage 1 entity-density rubric adapted for longer ideation method texts). Table 5 reports the mean and standard deviation per axis.

Table 5. Single-pass per-condition averages across n=30 problems, mean $\pm$ std. Validity and method specificity saturate near ceiling; coherence shows the C $\approx$ D parity that mirrors the pairwise judgment in Section 3.4.

Condition	Novelty	Validity	Coherence	Method spec.
A (no retrieval)	3.97 ± 0.41	5.00 ± 0.00	3.63 ± 0.49	4.40 ± 0.50
B (same-domain)	4.07 ± 0.25	4.97 ± 0.18	3.47 ± 0.51	4.40 ± 0.50
C (cross-domain)	4.13 ± 0.43	4.97 ± 0.18	3.33 ± 0.48	4.27 ± 0.45
D (random control)	4.13 ± 0.43	4.97 ± 0.18	3.33 ± 0.48	4.40 ± 0.50

C-centered pairwise assessments. Pairwise comparisons follow the same single-pass methods and the rubric-driven judge (Section 3.4); positions are randomized per call. Pairwise coherence is reported only for C vs D; per-condition coherence covers all four conditions in Table 5.

Table 6. Single-pass pairwise assessments on C-centered pairs, n=30 per pair. "Other" refers to the second condition in each pair (A, B, or D respectively). Bold marks the largest count in each row (winner or majority tie). Binomial p computed via Eq. 2. Pairwise validity ties account for 30–43% of comparisons, reinforcing the ceiling pattern observed in the per-condition table.

Pair	Axis	C wins	Other wins	Ties	p
C vs A	Novelty	**18**	12	0	0.36
	Validity	6	**15**	9	0.08
C vs B	Novelty	**20**	9	1	0.06
	Validity	9	8	**13**	1.00
C vs D	Novelty	14	**16**	0	0.86
	Validity	**12**	8	10	0.50
	Coherence	**17**	13	0	0.58

Table 7. Wilson 95% binomial confidence intervals for the C win rates in Figure 2. Intervals use all 30 problems. Ties are retained in the denominator, matching the displayed win-rate scale.

Pair	C wins / n	C win rate	95% CI
C vs A	18/30	60.0%	[42.3, 75.4]%
C vs B	20/30	66.7%	[48.8, 80.8]%
C vs D	14/30	46.7%	[30.2, 63.9]%

Attributed-seed incorporation. A held-out judge scored each attributed seed on a 1–3 scale (no / partial / full incorporation), applied to the unique seeds attributed under C (n=187) and D (n=191). Average integration is comparable (C 2.79, D 2.73). The discordant-pair breakdown for the McNemar test in Section 3.4 is: 7 problems show at least one unused attribution under D but not under C, versus 0 in the reverse direction; the paired test rejects equality at $p \approx 0.016$. Cross-domain retrieval thus reduces the rate of completely unused attributions but does not raise the typical level of seed integration.

F. Details

Models. GPT-5 (`openai.gpt-5`) generates all PaperGym outputs (seed extraction, paraphrasing, synthesis); Claude Sonnet 4.6 (`anthropic.claude-sonnet-4-6`) serves as the held-out judge for novelty, validity, coherence, and attributed-seed grounding. Embeddings use `text-embedding-3-small`.

Sandbox isolation. The Stage 1 accumulator runs each paper inside a Docker container.

Seed library. The retrieval library contains 1,167 seeds from 446 papers across the seven domains (see Section 3.1). Papers were sampled from Semantic Scholar with publication year in [2017, 2025] and a venue allowlist of major ML/CV/NLP/IR/speech/robotics conferences (NeurIPS, ICML, ICLR, CVPR, ICCV, ECCV, ACL, AAAI, EMNLP, SIGIR, KDD, Interspeech, ICRA, IROS, CoRL, RecSys, ICASSP). Per-domain target budgets were 107/89/77/60/60/60/47 for LLM_NLP/MULTIMODAL/CV/RL/IR_REC/SPEECH/ROBOTICS respectively; the resulting 446 papers reflect the post-sampling yield after Semantic Scholar filtering and successful extraction.

Random seeds. The random-seed control (condition D) samples its 21-seed pool at `seed=0`; the pairwise judge's position-randomization RNG uses `seed=42`. Both are fixed in the released scripts.

Cost. An end-to-end re-run of the released pipeline under GPT-5 generation ($1.25/$10 per 1M input/output tokens) and Claude Sonnet 4.6 judging ($3/$15 per 1M input/output tokens) at standard pricing measures approximately $80 of API cost across ∼10 h of wall-clock. Stage 3 single-pass ideation (A/B/C/D $\times$ 30 problems, with novelty rated by a multi-round ReAct judge that issues Semantic Scholar queries) dominates at $45 / 4.9 h; the C-centered pairwise novelty/validity judges add $3.4 / 0.9 h, the C-vs-D coherence judge $0.3 / < 0.1 h, and the novelty-iteration loop (Appendix C) contributes $29 / 3.0 h. The Q23 single-trace walkthrough (Appendix A) was measured directly at 74 s and $0.07 for the C-condition, consistent with the per-problem extrapolation.

Per-probe deduplication. Within each retrieval probe, only the first seed per source paper is kept before the top-k cut; subsequent seeds from the same paper are skipped. This prevents a single paper from dominating a probe's slate when its seeds embed similarly.

Lens passing. Each retrieved seed is presented to the synthesizer alongside a *lens*: the text that was actually embedded to retrieve the seed (raw problem statement, domain-specific paraphrase, or a placeholder for the random-seed control). The per-condition lens text is in Table 8. For the random-seed control (D) the lens explicitly states the seeds were sampled uniformly, so any C-vs-D parity reflects synthesizer use of seed content rather than ignorance of the random-sampling procedure.

Table 8. Lens text per condition.

Condition	Seeds	Lens text
A	none	(none)
B	top-21 same-domain	raw problem statement
C	top-3 $\times$ 7 paraphrase domains	domain paraphrase per seed; raw problem for natural-domain slot
D	21 random from library	"(uniform random sample of the library)"