CREATIVITYSUITE: A COMPREHENSIVE FRAMEWORK FOR EVALUATING AND ENHANCING CREATIVITY IN LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Creativity is widely regarded as a general capability of intelligence, yet its evaluation in large language models (LLMs) remains fragmented across domains and often conflates effectiveness with originality. We operationalize the standard definition of creativity through response-level quality and novelty, together with task-level creative space. Controlled experiments demonstrate that separate evaluation of quality and novelty improves the consistency and reference agreement of quality judgments, and that expanding creative space substantially increases the proportion of high-quality, high-novelty responses. Building on this formulation, we introduce CREATIVITYSUITE, a creativity training and evaluation framework spanning seven domains, with 19,858 training samples, 21,220 evaluation samples, and six complementary evaluation methods at both domain and instance levels. Experiments show that in-domain supervised fine-tuning yields localized gains, whereas joint training across domains yields inconsistent transfer and tends to improve correctness at the expense of novelty. Moreover, the measured effects of training vary considerably across evaluators. These findings indicate that imitating high-quality responses is insufficient for general creativity, which requires explicit coordination of quality and novelty. Once permitted, we will release our data and code.

*Creativity is the ability to produce ideas or outcomes that are **contextually original** and **effective** (Runco & Jaeger, 2012).*

1 INTRODUCTION

Across psychology, cognitive science, and education, creativity is widely regarded as a general capability of intelligence, characterized by both originality and effectiveness (Guilford, 1967; Ward et al., 1999; Plucker et al., 2004). However, how to operationalize this consensus as a measurable and comparable capability of large language models remains unclear. Creativity manifests differently across domains: mathematics emphasizes non-standard reasoning (Chen et al., 2025a; Ye et al., 2025), coding alternative algorithms (Lu et al., 2025; Wang et al., 2026), science novel and testable ideas (Ruan et al., 2026), and real-world design feasible solutions under practical constraints (Shah et al., 2003; Sarkar & Chakrabarti, 2011). Consequently, strong performance in a single domain may reflect domain knowledge or task adaptation rather than creativity that generalizes across contexts. This raises a fundamental question:

How should general creativity be defined across heterogeneous domains?

Answering this question requires identifying domain-invariant constructs while preserving domain-specific semantics. Rather than redefining creativity, we operationalize its established criteria for LLMs (Plucker et al., 2004; Runco & Jaeger, 2012): *effectiveness* and *originality* are formulated as response-side *quality* and *novelty*, while task-side *creative space* captures their contextual dependence. Quality measures whether a response satisfies task requirements, novelty whether it meaningfully departs from conventional solutions, and creative space the extent to which a task permits diverse valid solutions. To balance evaluation consistency with task specificity, we further introduce domain-level criteria to standardize evaluation semantics within each domain and instance-level rubrics to capture task-specific constraints and objectives.

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.

This formulation clarifies *what* should be measured, but not *how* its related dimensions should be assessed. Quality and novelty may be correlated in model outputs, but whether they should be measured jointly remains unclear. This leads to another compelling research question:

How should general creativity be evaluated reliably?

Existing benchmarks vary in definitions, references, and scoring protocols, often collapsing quality and novelty into a holistic judgment (Si et al., 2025; Padmakumar et al., 2025; Ruan et al., 2026; Lu et al., 2026). However, empirical association does not imply that their measurement should be coupled. In joint evaluation, originality may bias quality judgments, while correctness, fluency, or presentation may influence novelty judgments, obscuring whether an output is genuinely effective, substantively novel, or merely appears creative.

We compare joint and separate evaluation through controlled experiments. As shown in Figure 1, separate evaluation improves the order consistency of quality judgments by 5.86 percentage points (pp) across 700 response pairs from seven domains. On CreativeMath (Ye et al., 2025), we hold the tasks, responses, references, and scoring criteria fixed. Separation improves agreement with reference labels by 2.43 pp. Both improvements have 95% confidence intervals above zero, with novelty also showing positive point estimates. These results demonstrate that separate evaluation reliably improves quality judgments while making both dimensions clearly distinguishable and auditable. Further experimental details and results are provided in Appendix A.1.

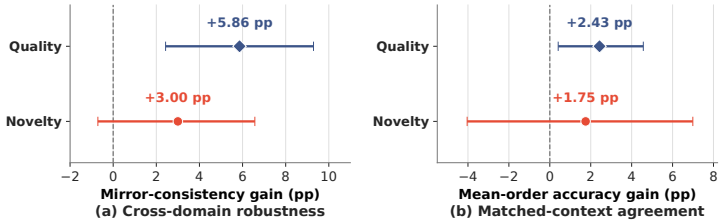

Figure 1: Empirical evidence for dimension-separated evaluation.

Towards this end, we introduce CREATIVITYSUITE, a framework for creativity training and evaluation across seven domains: mathematical reasoning, code generation, commonsense reasoning, physical problem solving, scientific ideation, creative writing, and real-world applications. It contains 19,858 training samples and 21,220 evaluation samples, together with six complementary evaluation methods spanning rule-based, LLM-based, and agent-based evaluators. The five model-based methods support both domain-level and instance-level scoring, while sharing the same model responses to isolate evaluation differences from generation variance.

We further validate creative space through cross-instance and within-instance controlled experiments. Expanding the space of valid solutions consistently increases novelty across three models and raises the proportion of high-quality, high-novelty responses by $32.26 \sim 46.51$ pp, with a modest decrease in quality. These results suggest that creative space captures opportunities for valid novel solutions while revealing a trade-off between creative opportunity and execution risk. Our experiments also show that Supervised Fine-Tuning (SFT) does not consistently improve general creativity. In-domain training typically yields more reliable localized gains, whereas joint training varies across domains, model scales, and evaluators, and may improve correctness or feasibility at the expense of novelty. This suggests that imitation of high-quality responses can introduce dimensional bias, and that future creativity training should explicitly coordinate quality and novelty, potentially through multi-objective reinforcement learning. Our main contributions are as follows:

- Through controlled experiments across seven domains, we show that the related dimensions of creativity should be evaluated separately. Separation significantly improves the robustness and reference agreement of quality judgments, providing empirical support for multidimensional creativity evaluation.
- We operationalize the standard definition of creativity through quality, novelty, and creative space, and validate the role of creative space in characterizing creative opportunities and calibrating value across tasks.
- We introduce CREATIVITYSUITE, spanning seven domains, six evaluation methods, two scoring modes, 19,858 training samples, and 21,220 evaluation samples, providing a comprehensive foundation for future research on creativity in large language models.

- We reveal unstable cross-domain transfer and quality bias under SFT, *i.e.*, training may improve effectiveness while constraining novelty, indicating that general creativity requires explicit coordination of multiple learning objectives.

2 The Creativity Benchmark

Existing studies operationalize LLM creativity within individual domains, which we group into seven domains, with some benchmarks spanning multiple domains.

- **Mathematical reasoning.** CreativeMath (Ye et al., 2025) evaluates whether models can produce correct solutions that differ from provided examples and collected reference solutions. DeepMath-Creative (Chen et al., 2025a) instead focuses on constructive proof and counterexamples. Both extend mathematical evaluation from answer correctness to valid alternative reasoning.

- **Code generation.** NeoCoder (Lu et al., 2025) progressively excludes techniques used by prior solutions, with its NeoGauge metric measuring correctness, constraint satisfaction, and divergence from human strategies. CreativeBench (Wang et al., 2026) evaluates combinatorial and exploratory creativity using executable correctness and solution novelty. These benchmarks require alternative implementations to remain functionally valid.

- **Commonsense reasoning.** CresOWLve (Ismayilzada et al., 2026) uses knowledge-grounded puzzles that require lateral connections, analogy, and abstraction. It evaluates answers through exact matching and LLM judgment, exposing the gap between retrieving relevant knowledge and combining it creatively.

- **Physical problem solving.** MacGyver (Tian et al., 2024) studies feasible problem solving with limited resources and unconventional object use. CreativityBench (Qian et al.) evaluates whether models select appropriate objects and parts and explain how their physical affordances enable new uses. Both benchmarks study creative tool use through text-based scenarios.

- **Scientific ideation.** LiveIdeaBench (Ruan et al., 2026) evaluates ideas generated from minimal scientific prompts along originality, feasibility, clarity, fluency, and flexibility. AI Idea Bench 2025 (Qiu et al., 2025) grounds idea evaluation in research papers and assesses content alignment, novelty, and feasibility. Relatedly, ScienceAgentBench (Chen et al., 2025b) evaluate executable scientific workflows, but focus on implementation success rather than idea novelty.

- **Creative writing.** CS4 (Atmakuru et al., 2024) varies prompt specificity to evaluate constraint satisfaction and narrative coherence. CreataSet (Cao et al., 2026) supports context-aware pairwise evaluation through its associated creativity evaluator, CrEval, while LitBench (Fein et al., 2026) benchmarks writing judges against debiased human preferences. EQ-Bench Creative Writing v3 (Paech, 2025) combines rubric-based scoring with pairwise comparisons to rank models. CreativityPrism (Hou et al., 2026b) situates creative writing alongside divergent thinking and logical reasoning, organizing task-specific metrics into quality, novelty, and diversity. These works illustrate the diversity of operational criteria used for creative text.

- **Real-world applications.** The marketing Creativity Benchmark (Bhat et al., 2025) evaluates brand-specific outputs using pairwise judgments from creative professionals and analyzes both within- and cross-model diversity. However, such evaluations remain application-specific and do not broadly cover creative problem solving under functional, resource, and deployment constraints, and has not publicly released its data or evaluation pipeline.

Cross-domain efforts take complementary approaches. NoveltyBench (Zhang et al., 2025) combines functional diversity across repeated generations with response quality, while AGC-Bench (Beaty et al., 2026) standardizes diverse benchmarks and calibrates LLM judges while retaining their original metrics. In contrast, CreativitySuite evaluates seven domains under a shared formulation of quality, novelty, and task-level creative space. Six complementary methods evaluate identical model responses, enabling controlled analysis of evaluator dependence. The accompanying training data further reveals whether creativity gains transfer across domains or favor quality at the expense of novelty.

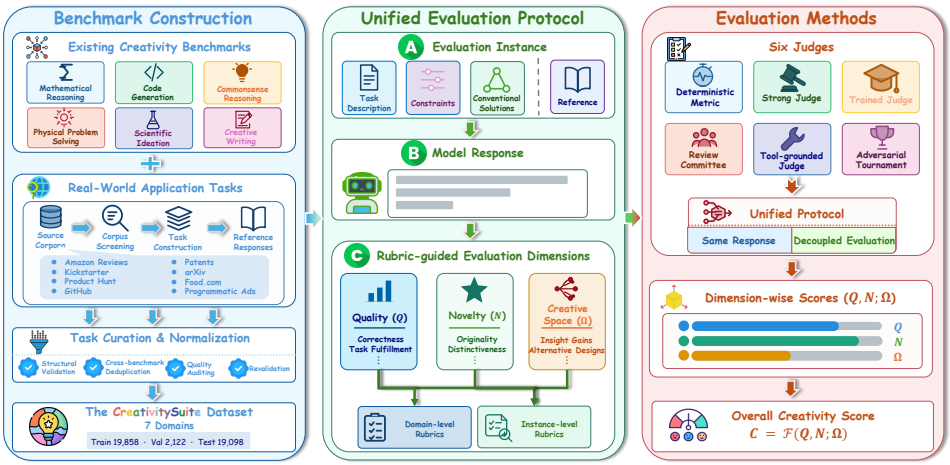

Figure 2: Overview of the CREATIVITYSUITE framework.

3 CREATIVITYSUITE

Figure 2 presents an overview of CREATIVITYSUITE. The framework comprises benchmark construction across seven creativity domains, a unified protocol over quality, novelty, and creative space, and six complementary evaluation methods.

3.1 FORMULATION OF CREATIVITY

Let $\mathcal{D}$ denote the set of creativity domains and $\mathcal{T}_d$ the tasks in domain d that admit substantially different yet valid responses. We represent each task $\tau_d \in \mathcal{T}_d$ as

$$\tau_d = (x, \mathcal{C}_x, \mathcal{R}_x; \Omega_x), \tag{1}$$

where x is the task description, $\mathcal{C}_x$ its constraints, $\mathcal{R}_x$ the conventional solution families, and Ω_x its creative space. For a response $y = f_\theta(x)$, we characterize creativity through response-side quality and novelty and task-side creative space:

$$\begin{cases} Q_d(y \mid \tau_d) &= \sum_{c \in \mathcal{C}_x} \alpha_{d,c} \sigma_{d,c}(y), \\ N_d(y \mid \tau_d) &= \sum_{a \in \mathcal{A}_d} \gamma_{d,a} \left[1 - \overline{S}^{(K)}_{d,a}(y, \mathcal{R}_x)\right], \\ \Omega_d(\tau_d) &= \sum_{b \in \mathcal{B}_d} \eta_{d,b} \omega_{d,b}(\tau_d), \end{cases} \tag{2}$$

where $\sigma_{d,c}$ measures constraint satisfaction, $\overline{S}^{(K)}_{d,a}$ is the mean similarity to the K nearest conventional solution families along novelty dimension a, and $\omega_{d,b}$ measures the opportunity for distinct valid solutions and meaningful gains. All scores lie in $[0, 1]$, and the weights within each dimension are non-negative and sum to one. The task-conditional creativity score is

$$C_d(y \mid \tau_d) = \mathcal{F}_d(Q_d(y \mid \tau_d), N_d(y \mid \tau_d); \Omega_d(\tau_d)), \tag{3}$$

where $\mathcal{F}_d$ is non-decreasing in each argument and satisfies

$$\mathcal{F}_d(0, N; \Omega) = \mathcal{F}_d(Q, 0; \Omega) = \mathcal{F}_d(Q, N; 0) = 0. \tag{4}$$

Thus, creativity requires an effective and substantively novel response to a task with non-zero creative space. Detailed dimension definitions, reference construction, and aggregation properties are provided in Appendix A.2.

3.2 THE CREATIVITYSUITE DATASET

3.2.1 TASK SELECTION

Domain Overview We partition the creativity tasks $\mathcal{T} = \bigsqcup_{d \in \mathcal{D}} \mathcal{T}_d$ into seven domains. Because existing benchmarks operationalize creativity differently across domains, we align all seven under the shared dimensions of quality, novelty, and creative space (Runco & Jaeger, 2012; Corazza, 2016). Detailed domain semantics, benchmark mappings, and representative tasks are provided in Tables 6 and 7 in Appendix A.3.

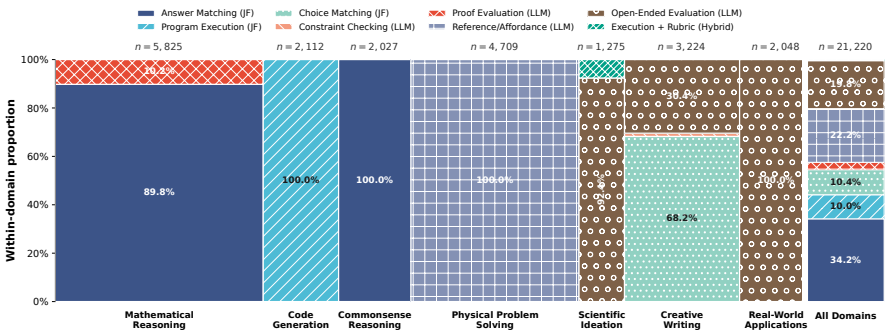

Figure 4: Verification-route distributions across the seven CREATIVITYSUITE domains. JF denotes judge-free evaluation.

Real-World Application Tasks. Existing benchmarks rarely examine whether models can develop novel and feasible solutions under practical constraints. We therefore construct real-world application tasks spanning product, software, research, culinary, and service design. Candidate records from eight grounded corpora undergo cleaning, stratified sampling, semantic deduplication, and quality filtering before being transformed into self-contained tasks through *reconstruction* or *synthesis*. Each task is represented as $\tau_d = (x, \mathcal{C}_x, \mathcal{R}_x; \Omega_x)$, and training responses are selected from independently generated candidates through rubric-based evaluation. The task taxonomy, corpus sources, construction routes, screening procedures, and response selection are detailed in Appendices A.4–A.8; Tables 8 and 9 summarize the corpora and construction routes.

3.2.2 TASK CURATION

We harmonize twelve public benchmarks across six domains and combine them with our real-world application tasks. Evaluation data preserve source semantics and references while undergoing structural validation, cross-source deduplication, and quality auditing. Training data prioritize source-provided samples and supplement underrepresented task types through independently constructed tasks and responses, followed by rejection sampling. Full curation and verification procedures are provided in Appendix A.9.



Figure 3: Knowledge-theme distributions across the seven CREATIVITYSUITE domains.

Data Statistics. CREATIVITYSUITE contains 19,858 training and 21,220 evaluation instances, including 2,122 for validation and 19,098 for testing. Figures 3 and 4 show its knowledge-theme coverage and verification-route composition, respectively.

3.3 EVALUATION PROTOCOL

To support cross-domain comparison, CREATIVITYSUITE separately evaluates response-side quality and novelty, together with task-side creative space.

3.3.1 EVALUATION DIMENSIONS

For response y_i to task τ_i from domain d_i, we define

$$Q_i = 10Q_{d_i}(y_i \mid \tau_i), \qquad N_i = 10N_{d_i}(y_i \mid \tau_i), \qquad \Omega_i = 10\Omega_{d_i}(\tau_i), \tag{5}$$

where all scores lie in $[0, 10]$. Creative space Ω_i is determined before response generation and remains fixed across models and evaluators.

Scoring Modes. The *domain-level* mode applies shared criteria within each domain, whereas the *instance-level* mode uses task-specific rubrics:

$$\begin{cases} E_{i,k}^{\text{domain}} & = \frac{1}{|\mathcal{U}_{d_i,k}|} \sum_{u \in \mathcal{U}_{d_i,k}} e_{i,k,u}, \\ E_{i,k}^{\text{instance}} & = 10 \frac{\sum_{j \in \mathcal{J}_{i,k}} r_{i,k,j}}{\sum_{j \in \mathcal{J}_{i,k}} r_{i,k,j}^{\max}}, \qquad k \in \{Q, N\}. \end{cases} \tag{6}$$

Quality and novelty are evaluated independently. For model-based evaluation with an objective verifier $v_i \in [0, 1]$, quality is

$$E_{i,Q}^{(m)} = 0.6(10v_i) + 0.4E_{i,Q}^{(m,\text{rem})}, \tag{7}$$

where the second term captures quality beyond objective correctness.

Score Aggregation. The three dimensions are combined using a domain-weighted geometric mean:

$$C_i^{(m)} = 10 \left(\frac{E_{i,Q}^{(m)}}{10} \right)^{w_{d_i,Q}} \left(\frac{E_{i,N}^{(m)}}{10} \right)^{w_{d_i,N}} \left(\frac{\Omega_i}{10} \right)^{w_{d_i,\Omega}}, \tag{8}$$

where the positive weights sum to one and $m \in \{\text{metric}, \text{domain}, \text{instance}\}$. Thus, a low value on any dimension limits overall creativity. Full scoring definitions and weights are provided in Appendix A.10.

3.3.2 Evaluation Methods

We implement one deterministic metric and five model-based evaluators. All methods reuse the same stored responses and frozen evaluation artifacts, separating evaluator variation from generation variation. The model-based methods support both scoring modes and evaluate quality and novelty through independent calls.

Deterministic Metric. *Metric* requires no online model judgment. When an objective verifier is available, $E_{i,Q}^{\text{metric}} = 10v_i$; otherwise, quality is estimated against the frozen anchor bank $\mathcal{Q}_{x_i}$:

$$E_{i,Q}^{\text{metric}} = 10 \ \text{clip}\left(\alpha_{d_i} \max_{a \in \mathcal{Q}_{x_i}} [0.4 f_{\text{sem}}(y_i, a) + 0.6 f_{\text{cov}}(y_i, a)] + \beta_{d_i}, 0, 1 \right). \tag{9}$$

For novelty, response units $s \in \mathcal{S}_i$ and conventional mechanism atoms $p \in \mathcal{P}(\mathcal{R}_{x_i})$ are projected away from the task direction $\mathbf{h}_i$:

$$\begin{cases} \widetilde{\mathbf{u}}_s & = \text{norm}(\mathbf{u}_s - \langle \mathbf{u}_s, \mathbf{h}_i \rangle \mathbf{h}_i), \\ \widetilde{\mathbf{v}}_p & = \text{norm}(\mathbf{v}_p - \langle \mathbf{v}_p, \mathbf{h}_i \rangle \mathbf{h}_i). \end{cases} \tag{10}$$

Their calibrated conventionality distance d_s is weighted by task relevance ρ_s, giving $z_s = d_s \rho_s^2$. The final score averages the two strongest mechanism units:

$$\begin{cases} d_s & = 1 - \text{clip}\left(\frac{\max_{p \in \mathcal{P}(\mathcal{R}_{x_i})} \langle \widetilde{\mathbf{u}}_s, \widetilde{\mathbf{v}}_p \rangle - \ell_{\text{conv}}}{u_{\text{conv}} - \ell_{\text{conv}}}, 0, 1 \right), \\ E_{i,N}^{\text{metric}} & = \frac{10}{K_i} \sum_{s \in \text{TopK}_{K_i}(\mathcal{S}_i; d_s \rho_s^2)} d_s \rho_s^2, \qquad K_i = \min(2, |\mathcal{S}_i|). \end{cases} \tag{11}$$

Task relevance suppresses unrelated content, while Top-K aggregation limits the effect of anomalous segments. Full definitions and calibration details are provided in Appendix A.10.5.

Model-Based Methods. For dimension $k \in \{Q, N\}$ and mode $m \in \{\text{domain}, \text{instance}\}$, each evaluator receives

$$\xi_{i,k}^{(m)} = \left(\tau_i, y_i, \mathcal{A}_{i,k}, \Pi_{i,k}^{(m)} \right), \tag{12}$$

where $\Pi_{i,k}^{(m)}$ contains only the criteria for dimension k. $\mathcal{A}_{i,Q}$ provides verifier evidence, whereas $\mathcal{A}_{i,N}$ provides conventional-solution references; information from the other dimension is withheld.

The *Strong Judge* directly applies a frontier evaluator, and the *Trained Judge* provides an efficient evaluator trained on criterion-level supervision. The *Review Committee* resolves independent reviews through deliberation; the *Tool-Grounded Judge* incorporates retrieved or executed evidence; and the *Adversarial Tournament* tests responses through challenger–defender debate. Detailed procedures and prompts are provided in Appendices A.10 and A.10.9.

Table 1: Results on six domain-specific creativity benchmarks. SFT denotes in-domain SFT, while SFT* denotes SFT on all seven domains. ***Avg.*** is the macro-average over the six benchmarks after mapping all metrics to $[0, 100]$ and averaging the displayed metrics within each benchmark. **Best results** within each model scale are shown in bold. Results for the Gemma models are provided in the Appendix A.13.

Model	Method	CreativeMath		CreativeBench		CresOWLve	CreativityBench		LiveIdeaBench	CreataSet	*Avg.*
		Correctness ↑	Novelty ↑	Pass@1 ↑	Creativity ↑	Accuracy ↑	Entity Correct ↑	Gold Correct ↑	Average ↑	Accuracy ↑	
Qwen3.5-4B	Base	34.38	10.11	24.85	0.1639	**2.97**	66.46	16.55	6.2455	73.23	37.17
	SFT	**43.57**	**15.26**	**25.51**	**0.1655**	2.64	**78.15**	15.37	**6.4981**	73.68	**39.75**
	SFT*	40.26	8.27	24.49	0.1584	2.70	68.90	**17.12**	5.9630	**74.52**	37.38
Qwen3.5-9B	Base	36.95	15.07	28.98	0.1891	3.88	62.11	19.22	6.3510	73.35	38.56
	SFT	**52.02**	**15.99**	**29.82**	**0.1925**	**4.42**	**73.66**	**20.30**	**6.6265**	**73.76**	**41.66**
	SFT*	40.63	11.76	28.38	0.1808	3.77	62.92	17.09	6.4260	73.31	38.46
Qwen3.8-27B	Base	38.42	16.36	50.24	0.3311	6.09	48.19	19.93	**6.5829**	**72.38**	41.24
	SFT	**41.54**	**19.30**	50.54	0.3342	**6.15**	**55.15**	**23.54**	6.3427	72.22	**42.26**
	SFT*	39.52	13.60	**50.66**	**0.3363**	**6.15**	50.05	20.57	6.4334	71.45	40.99

4 Empirical Studies

4.1 Training Settings

Models. We evaluate two model families across multiple scales: Qwen3.5-4B, Qwen3.5-9B, and Qwen3.8-27B (Qwen Team, 2026a;b), as well as Gemma-4-12B and Gemma-4-31B (Team, 2026b).

Training. We conduct single-domain SFT on six domain-specific datasets and seven-domain SFT on their mixed training data. The former measures the contribution of domain-specific data, whereas the latter evaluates joint training and cross-domain transfer. All experiments use a unified training framework, with detailed configurations provided in the Appendix A.12. The Appendix A.11 also reports additional experiments validating the effectiveness of the real-world applications data.

4.2 Evaluation Benchmarks

Domain-Specific Creativity. We evaluate mathematics, code, commonsense, physical reasoning, science, and writing using CreativeMath (Ye et al., 2025), CreativeBench (Wang et al., 2026), CresOWLve (Ismayilzada et al., 2026), CreativityBench (Qian et al.), LiveIdeaBench (Ruan et al., 2026), and CreataSet (Cao et al., 2026), respectively. Comparisons against the corresponding base models quantify the effects of single-domain and seven-domain SFT.

Cross-Domain Creativity. We evaluate base and seven-domain SFT models on the CreativitySuite test set using six evaluation methods. Five model-based methods operate at both the domain and instance levels, while the Deterministic Metric uses a fixed scoring protocol. All applicable methods evaluate the same model responses to control for generation variance.

Agentic Generalization. We further evaluate base and seven-domain SFT models on DeepResearch-Benchmarks (Team, 2026a), sampling up to 100 questions per benchmark, to assess transfer to complex agentic research tasks. We follow the default agent, inference, and evaluation settings.

4.3 Main Results

4.3.1 Domain-Specific Creativity Evaluation

Table 1 compares the base, in-domain SFT, and full-domain SFT (SFT*) models under the original evaluation protocols of six domain-specific benchmarks. The key findings are as follows:

- **In-domain supervision strengthens the corresponding creative abilities, but the gains diminish with increasing model scale.** Mathematical reasoning and physics problem solving benefit most, showing that domain supervision effectively enhances creative problem solving. In contrast, gains in scientific ideation and creative writing are less consistent, suggesting that open-ended generation is harder to improve reliably through domain supervision.

Table 2: Cross-domain creativity evaluation on CREATIVITYSUITE. Each entry reports the SFT⋆ creativity score, with the corresponding Base score shown as a subscript. Within each entry, the better score is shown in bold and the worse score in gray. Entries unchanged after rounding to two decimal places are shown in regular type. These styles describe score direction and do not imply statistical significance. ***Avg.*** is the macro-average over the seven domains. All methods evaluate the same frozen responses, and all scores are rounded to two decimal places. Detailed results are reported in the Appendix A.15.

Evaluator	Model	Math	Code	Common.	Physical	Science	Writing	RealWorld	*Avg.*
Deterministic Metric	4B	**1.53**$_{1.39}$	**7.73**$_{7.42}$	0.08$_{\mathbf{0.09}}$	**11.97**$_{11.93}$	**15.56**$_{14.56}$	**10.77**$_{10.36}$	**27.71**$_{26.77}$	**10.77**$_{10.36}$
	9B	**1.78**$_{1.33}$	**9.32**$_{9.05}$	0.13$_{\mathbf{0.18}}$	**10.96**$_{10.48}$	**14.47**$_{14.09}$	**10.74**$_{10.64}$	**27.90**$_{26.71}$	**10.76**$_{10.36}$
	27B	**2.60**$_{2.21}$	15.14$_{\mathbf{15.32}}$	0.11$_{0.11}$	**7.63**$_{7.49}$	12.97$_{\mathbf{13.49}}$	**11.40**$_{11.33}$	**27.78**$_{27.64}$	**11.09**$_{11.08}$
Domain-level evaluation									
Strong Judge	4B	5.40$_{\mathbf{5.53}}$	11.93$_{\mathbf{17.02}}$	1.12$_{\mathbf{1.39}}$	24.06$_{\mathbf{32.78}}$	**41.25**$_{36.81}$	57.76$_{\mathbf{58.32}}$	58.15$_{\mathbf{58.31}}$	28.52$_{\mathbf{30.02}}$
	9B	7.25$_{\mathbf{7.47}}$	12.23$_{\mathbf{18.44}}$	1.97$_{\mathbf{2.27}}$	25.39$_{\mathbf{32.03}}$	**44.82**$_{37.90}$	59.35$_{\mathbf{61.20}}$	60.80$_{\mathbf{60.85}}$	30.26$_{\mathbf{31.45}}$
	27B	**7.46**$_{7.30}$	12.95$_{\mathbf{24.29}}$	**3.21**$_{2.89}$	**34.49**$_{30.06}$	**48.33**$_{41.90}$	61.91$_{\mathbf{62.13}}$	60.89$_{\mathbf{62.66}}$	32.75$_{\mathbf{33.03}}$
Trained Judge	4B	**22.03**$_{21.19}$	19.97$_{\mathbf{23.13}}$	17.49$_{\mathbf{17.78}}$	**20.47**$_{17.32}$	31.69$_{\mathbf{35.19}}$	32.01$_{\mathbf{37.35}}$	**34.95**$_{33.74}$	25.52$_{\mathbf{26.53}}$
	9B	**21.98**$_{21.25}$	21.23$_{\mathbf{21.76}}$	17.25$_{\mathbf{18.26}}$	**20.76**$_{17.00}$	**33.97**$_{32.79}$	32.29$_{\mathbf{36.99}}$	**39.39**$_{29.30}$	**26.70**$_{25.33}$
	27B	**21.96**$_{21.18}$	21.25$_{\mathbf{21.63}}$	**18.05**$_{17.39}$	**20.34**$_{17.50}$	32.69$_{\mathbf{34.25}}$	33.13$_{\mathbf{36.07}}$	**43.72**$_{25.22}$	**27.30**$_{24.75}$
Review Committee	4B	**2.95**$_{2.79}$	4.84$_{\mathbf{11.32}}$	0.72$_{\mathbf{0.73}}$	11.34$_{\mathbf{32.94}}$	54.10$_{\mathbf{54.47}}$	54.57$_{\mathbf{54.71}}$	**71.54**$_{71.53}$	28.58$_{\mathbf{32.64}}$
	9B	**3.28**$_{2.46}$	6.46$_{\mathbf{13.17}}$	1.18$_{\mathbf{1.21}}$	**32.26**$_{31.39}$	**57.61**$_{56.00}$	57.70$_{\mathbf{57.92}}$	**73.64**$_{73.44}$	33.16$_{\mathbf{33.66}}$
	27B	**3.66**$_{3.05}$	9.48$_{\mathbf{22.93}}$	1.86$_{\mathbf{1.88}}$	15.31$_{\mathbf{25.63}}$	58.65$_{\mathbf{59.77}}$	59.60$_{\mathbf{59.94}}$	**74.74**$_{74.69}$	31.90$_{\mathbf{35.41}}$
Tool-Grounded Judge	4B	**3.67**$_{3.17}$	0.00$_{\mathbf{0.03}}$	8.21$_{\mathbf{8.60}}$	10.86$_{\mathbf{32.34}}$	47.08$_{\mathbf{47.29}}$	46.25$_{\mathbf{46.69}}$	**63.49**$_{63.48}$	25.65$_{\mathbf{28.80}}$
	9B	**3.59**$_{3.02}$	0.00$_{0.00}$	10.55$_{\mathbf{10.84}}$	10.85$_{\mathbf{30.64}}$	**49.96**$_{48.50}$	49.72$_{\mathbf{50.24}}$	**66.20**$_{65.95}$	27.27$_{\mathbf{29.88}}$
	27B	**3.89**$_{3.47}$	0.03$_{\mathbf{0.12}}$	**10.26**$_{10.11}$	14.38$_{\mathbf{24.67}}$	**55.29**$_{54.00}$	50.69$_{\mathbf{51.27}}$	67.79$_{\mathbf{67.95}}$	28.90$_{\mathbf{30.23}}$
Adversarial Tournament	4B	**5.76**$_{4.37}$	**13.65**$_{13.36}$	18.44$_{\mathbf{18.83}}$	**27.10**$_{26.35}$	34.98$_{\mathbf{36.61}}$	42.90$_{\mathbf{43.20}}$	63.82$_{\mathbf{64.40}}$	29.52$_{\mathbf{29.59}}$
	9B	**5.13**$_{4.05}$	**16.04**$_{15.89}$	21.91$_{\mathbf{22.36}}$	**26.42**$_{26.41}$	**37.69**$_{37.44}$	44.55$_{\mathbf{46.29}}$	66.39$_{\mathbf{66.48}}$	31.16$_{\mathbf{31.28}}$
	27B	**5.54**$_{4.99}$	27.77$_{\mathbf{27.83}}$	**24.29**$_{20.81}$	**23.71**$_{23.17}$	**41.86**$_{41.66}$	47.01$_{\mathbf{47.38}}$	67.30$_{\mathbf{67.76}}$	**33.93**$_{33.37}$
Instance-level evaluation									
Strong Judge	4B	**16.69**$_{16.01}$	21.02$_{\mathbf{27.14}}$	3.86$_{\mathbf{4.44}}$	25.33$_{\mathbf{35.49}}$	41.19$_{\mathbf{43.90}}$	64.65$_{\mathbf{65.62}}$	**68.95**$_{68.62}$	34.53$_{\mathbf{37.32}}$
	9B	19.08$_{\mathbf{19.20}}$	22.69$_{\mathbf{30.98}}$	5.69$_{\mathbf{5.91}}$	27.63$_{\mathbf{37.17}}$	44.77$_{\mathbf{44.82}}$	66.76$_{\mathbf{69.89}}$	**72.88**$_{72.27}$	37.07$_{\mathbf{40.03}}$
	27B	18.78$_{\mathbf{18.81}}$	24.37$_{\mathbf{37.53}}$	**7.36**$_{7.05}$	37.72$_{\mathbf{39.19}}$	**49.75**$_{49.25}$	71.44$_{\mathbf{71.74}}$	75.17$_{\mathbf{76.25}}$	40.66$_{\mathbf{42.83}}$
Trained Judge	4B	**21.86**$_{21.30}$	20.10$_{\mathbf{22.90}}$	17.53$_{\mathbf{17.65}}$	18.64$_{\mathbf{18.88}}$	31.98$_{\mathbf{34.84}}$	31.84$_{\mathbf{37.46}}$	31.73$_{\mathbf{37.11}}$	24.81$_{\mathbf{27.16}}$
	9B	**22.02**$_{21.12}$	20.94$_{\mathbf{22.01}}$	17.38$_{\mathbf{17.98}}$	18.77$_{\mathbf{18.79}}$	**34.24**$_{32.54}$	32.29$_{\mathbf{36.88}}$	**36.07**$_{32.70}$	25.96$_{\mathbf{26.00}}$
	27B	**21.99**$_{21.11}$	21.38$_{\mathbf{21.45}}$	17.29$_{\mathbf{17.92}}$	**19.53**$_{18.11}$	33.29$_{\mathbf{33.52}}$	33.19$_{\mathbf{35.98}}$	**42.55**$_{26.38}$	**27.03**$_{24.93}$
Review Committee	4B	**3.23**$_{3.14}$	4.74$_{\mathbf{13.14}}$	**0.21**$_{0.17}$	12.37$_{\mathbf{32.75}}$	53.12$_{\mathbf{57.07}}$	55.94$_{\mathbf{56.23}}$	75.99$_{\mathbf{76.21}}$	29.37$_{\mathbf{34.10}}$
	9B	**3.44**$_{2.79}$	6.79$_{\mathbf{15.83}}$	**0.39**$_{0.38}$	**33.04**$_{32.06}$	57.64$_{\mathbf{58.16}}$	59.88$_{\mathbf{60.43}}$	78.41$_{\mathbf{78.57}}$	34.23$_{\mathbf{35.46}}$
	27B	**3.82**$_{3.40}$	9.29$_{\mathbf{26.83}}$	0.54$_{\mathbf{0.58}}$	18.15$_{\mathbf{29.14}}$	59.72$_{\mathbf{62.05}}$	62.13$_{\mathbf{62.87}}$	80.00$_{\mathbf{80.12}}$	33.38$_{\mathbf{37.85}}$
Tool-Grounded Judge	4B	**4.60**$_{3.86}$	0.00$_{0.00}$	2.27$_{\mathbf{2.34}}$	11.72$_{\mathbf{32.72}}$	43.72$_{\mathbf{45.39}}$	46.44$_{\mathbf{47.04}}$	67.38$_{\mathbf{67.72}}$	25.16$_{\mathbf{28.44}}$
	9B	**4.47**$_{3.71}$	**0.05**$_{0.00}$	3.13$_{\mathbf{3.19}}$	12.35$_{\mathbf{32.20}}$	**46.61**$_{45.19}$	51.72$_{\mathbf{52.42}}$	**71.43**$_{71.38}$	27.11$_{\mathbf{29.73}}$
	27B	**5.02**$_{4.32}$	0.00$_{0.00}$	**3.85**$_{3.84}$	16.96$_{\mathbf{28.00}}$	**53.60**$_{51.67}$	52.89$_{\mathbf{53.56}}$	73.73$_{\mathbf{74.09}}$	29.44$_{\mathbf{30.78}}$
Adversarial Tournament	4B	**4.96**$_{4.17}$	**14.19**$_{13.77}$	1.74$_{\mathbf{2.20}}$	**27.40**$_{26.80}$	35.21$_{\mathbf{38.48}}$	44.65$_{\mathbf{44.93}}$	67.72$_{\mathbf{67.98}}$	27.98$_{\mathbf{28.33}}$
	9B	**4.96**$_{3.88}$	15.66$_{\mathbf{16.58}}$	3.05$_{\mathbf{3.14}}$	27.32$_{\mathbf{27.36}}$	38.71$_{\mathbf{39.20}}$	45.79$_{\mathbf{48.15}}$	**71.33**$_{71.32}$	29.55$_{\mathbf{29.95}}$
	27B	**5.56**$_{4.87}$	27.84$_{\mathbf{28.02}}$	**3.28**$_{2.91}$	**25.55**$_{25.10}$	42.91$_{\mathbf{43.68}}$	49.82$_{\mathbf{50.66}}$	73.53$_{\mathbf{73.98}}$	32.64$_{\mathbf{32.75}}$

- **Full-domain SFT exhibits a clear quality–novelty trade-off.** On math tasks, correctness generally improves while novelty declines, indicating that joint training makes models more reliable but biases them toward conventional solutions. Gains in other domains also vary across model scales and evaluation dimensions, with no consistent cross-domain improvement.
- **In-domain SFT outperforms full-domain SFT.** Highly relevant domain supervision translates more readily into stable gains. Full-domain training has yet to achieve broadly positive transfer, with cross-domain interference remaining a key challenge for stable creativity transfer.

4.3.2 CROSS-DOMAIN CREATIVITY EVALUATION

Table 2 compares the overall creativity scores of the Base and SFT⋆ models across different evaluation methods. Figures 26 and 27 in Appendix A.15 further show the changes in quality and novelty. The main findings are as follows:

- **Evaluation conclusions are sensitive to the evaluation method.** The rule-based evaluator (*i.e.*, the deterministic metric) suggests a modest improvement from SFT⋆, whereas generative evaluators generally indicate a decline. The discriminative evaluator (*i.e.*, the trained judge) becomes

Table 3: Results on Deep-Research-Benchmarks under the corresponding recorded generation settings. SFT⋆ denotes SFT on all seven creativity domains. Accuracy (Acc.), Exact Match (EM), F1, Structured F1 (S-F1), and Table Completion (TC) are reported as percentages; RS denotes Rubric Score and retains its original scale. All benchmarks contain 100 evaluated instances except FrontierScience-Research, which contains 60. For WideSearch, S-F1 is macro-averaged over all instances, while TC denotes the percentage of responses from which the frozen scorer extracts a valid table. Best results within each model scale are shown in bold.

Model	Setting	Browse Comp	DeepSearch QA		XBench-DeepResearch	FrontierScience-Research		SUPERChem-Text	WideSearch			FrontierScience-Olympiad
		Acc. ↑	EM ↑	F1 ↑	Acc. ↑	Acc. ↑	RS ↑	Acc. ↑	EM ↑	S-F1 ↑	TC ↑	Acc. ↑
Qwen3.5-4B	Base	**31.00**	**47.00**	**62.80**	31.00	3.33	2.614	10.00	1.00	**57.03**	**99.00**	34.00
	SFT⋆	29.00	43.00	59.75	**36.00**	**10.00**	**2.808**	**16.00**	**2.00**	39.16	71.00	**38.00**
Qwen3.5-9B	Base	33.00	48.00	66.11	36.00	**8.33**	**2.980**	17.00	1.00	**59.71**	**96.00**	39.00
	SFT⋆	**36.00**	**53.00**	**68.41**	**45.00**	**8.33**	2.583	**21.00**	**2.00**	49.68	78.00	**48.00**
Qwen3.8-27B	Base	**39.00**	66.00	84.90	48.00	**18.33**	**2.849**	9.00	2.00	**33.37**	**51.00**	49.00
	SFT⋆	33.00	**67.00**	**85.36**	**56.00**	**18.33**	2.792	**14.00**	**3.00**	31.69	47.00	**54.00**

more favorable as model scale increases. Taken together, these divergent trends reveal limited agreement among current creativity evaluation methods.

- **Instance-level evaluation is stricter than domain-level evaluation.** Under the same evaluator and model scale, instance-level evaluation typically yields more negative changes, indicating that fine-grained criteria better expose capability degradation masked by domain-level aggregation.
- **Different domains exhibit relatively consistent but distinct patterns of change.** Mathematical tasks generally improve in quality but decline in novelty. Writing tasks show declines in quality, novelty, and overall creativity, while coding tasks also suffer notable quality loss. Results in science, physics, and real-world applications vary across evaluators, with no consistent positive transfer.

It's noted that the composite creativity score is not a simple average of quality and novelty; it also depends on the creative space and weighting of individual instances. Moreover, CREATIVITYSUITE comprises refined data from multiple external benchmarks, and evaluator suitability may vary across domains. More detailed results and evaluation protocols are provided in the Appendix A.15.

4.3.3 AGENTIC GENERALIZATION EVALUATION

Table 3 shows selective transfer from seven-domain creativity training to agentic research tasks. All three models improve on XBench-DeepResearch, SUPERChem-Text, and FrontierScience-Olympiad, suggesting that creativity training can benefit some open-ended research and scientific problem-solving tasks. Qwen3.5-9B shows the most consistent transfer, improving or maintaining performance on nearly all available metrics. However, the gains are not universal: Qwen3.5-4B declines on BrowseComp and DeepSearchQA, while Qwen3.8-27B degrades on BrowseComp and WideSearch Structured F1. Thus, creativity training more consistently benefits research and science tasks, but does not reliably improve retrieval or structured information aggregation.

Additionally, Appendix A.14 provides further analysis of the effect of creative space and validation of the deterministic metric.

5 CONCLUSION AND FUTURE WORK

We present CREATIVITYSUITE, a framework for training and evaluating creativity across seven domains. CREATIVITYSUITE characterizes creativity through response quality, response novelty, and task-level creative space, and supports six complementary evaluation methods at both domain and instance levels. Our experiments show that quality and novelty should be evaluated separately, and that in-domain supervision yields more consistent gains than joint training, which may improve correctness at the expense of novelty. Future work will extend this framework beyond parametric knowledge to interactive settings with external environments.

AI USE STATEMENT

Generative AI tools are used in this work in three distinct roles. First, large language models (LLMs) are components of the proposed evaluation protocol itself: as described in Section 3.3, five of the six evaluation methods (Strong Judge, Trained Judge, Review Committee, Tool-Grounded Judge, and Adversarial Tournament) employ LLMs to score response quality and novelty, and an LLM judge adjudicates commonsense answers that appear in neither the accepted nor the rejected answer pool (Figure 4). This use is intrinsic to CREATIVITYSUITE and is reported together with the prompts in Appendix A.10.9. Second, we use LLMs to generate synthetic data. All real-world application tasks are constructed by GPT-5.5 from screened real-world corpora through the *Reconstruction* and *Synthesis* routes (Appendix A.7), and the supervision targets for their training tasks are selected through rubric-based evaluation from candidate responses generated by multiple independent frontier LLMs (Appendix A.8). For domains with insufficient training data, supplementary samples are generated by GPT-5.5 and Claude Opus 4.8 and filtered through rejection sampling that combines deterministic verification, independent LLM review, semantic deduplication, and human review of disputed cases (Section 3.2.2 and Appendix A.9). The Trained Judge is further trained on rubric-based supervision generated by the Strong Judge. In contrast, evaluation instances in the six existing domains are taken from their source benchmarks, where LLM-based review is used only for quality auditing in combination with native validators and human adjudication. Third, beyond these uses, we used generative AI tools as writing assistants, both to produce initial English drafts of parts of the paper from outlines and notes prepared by the authors, and to polish the text through grammar correction, style improvement, and wording adjustments for clarity. The research questions, formulation, experimental design, results, and conclusions are the authors' own, and all AI-drafted text was substantially revised by the authors. Generative AI tools were not used to formulate research ideas, to search for related work, or to prove mathematical claims. All AI-assisted text was reviewed and verified by the authors, all synthetic data was subjected to the verification procedures described above, and we take full responsibility for the final content of this work, including its text, claims, and artifacts.

REFERENCES

Anirudh Atmakuru, Jatin Nainani, Rohith Siddhartha Reddy Bheemreddy, Anirudh Lakkaraju, Zonghai Yao, Hamed Zamani, and Haw-Shiuan Chang. CS4: Measuring the creativity of large language models automatically by controlling the number of story-writing constraints. *arXiv preprint arXiv:2410.04197*, 2024.

Roger Beaty, Vijeta Deshpande, Clin KY Lai, Anna Attuch, Namrata Shivagunde, Swastik Roy, Rajkumar Pujari, Paul V DiStefano, Sherin Muckatira, Claire E Stevenson, et al. AGC-Bench: Measuring artificial general creativity. *arXiv preprint arXiv:2607.01152*, 2026.

Ninad Bhat, Kieran Browne, and Pip Bingemann. Creativity benchmark: A benchmark for marketing creativity for large language models. *arXiv preprint arXiv:2509.09702*, 2025.

Qian Cao, Xiting Wang, Yuzhuo Yuan, Yahui Liu, Fang Luo, and Ruihua Song. Evaluating text creativity across diverse domains: A dataset and large language model evaluator. In *International Conference on Learning Representations*, volume 2026, pp. 33245–33283, 2026.

Xiaoyang Chen, Xinan Dai, Yu Du, Qian Feng, Naixu Guo, Tingshuo Gu, Yuting Gao, Yingyi Gao, Xudong Han, Xiang Jiang, et al. DeepMath-Creative: A benchmark for evaluating mathematical creativity of large language models. *arXiv preprint arXiv:2505.08744*, 2025a.

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

Giovanni Emanuele Corazza. Potential originality and effectiveness: The dynamic definition of creativity. *Creativity research journal*, 28(3):258–267, 2016.

Daniel Fein, Sebastian Russo, Violet Xiang, Kabir Jolly, Rafael Rafailov, and Nick Haber. LitBench: A benchmark and dataset for reliable evaluation of creative writing. In *Proceedings of the 19th*

Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 7740–7755, 2026.

Joy P Guilford. Creativity: Yesterday, today and tomorrow. *The journal of creative behavior*, 1(1): 3–14, 1967.

Yupeng Hou, Jiacheng Li, Xiangjun Fu, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation: Benchmarking LLMs as semantic encoders. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3251–3265, 2026a.

Zhaoyi Joey Hou, Alvin Zhang, Yining Lu, Bhiman Kumar Baghel, Anneliese Brei, Ximing Lu, Meng Jiang, Faeze Brahman, Snigdha Chaturvedi, Haw-Shiuan Chang, Daniel Khashabi, and Xiang Lorraine Li. CreativityPrism: A cross-domain evaluation framework for large language model creativity. *Transactions on Machine Learning Research*, 2026b. ISSN 2835-8856.

Mete Ismayilzada, Renqing Cuomao, Daniil Yurshevich, Anna Sotnikova, Lonneke van der Plas, and Antoine Bosselut. CresOWLve: Benchmarking creative problem-solving over real-world knowledge. *arXiv preprint arXiv:2604.03374*, 2026.

Li-Chun Lu, Miri Liu, Pin Chun Lu, Yufei Tian, Shao-Hua Sun, and Nanyun Peng. Rethinking creativity evaluation: A critical analysis of existing creativity evaluations. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6329–6352, 2026.

Yining Lu, Dixuan Wang, Tianjian Li, Dongwei Jiang, Sanjeev Khudanpur, Meng Jiang, and Daniel Khashabi. Benchmarking language model creativity: A case study on code generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 2776–2794, 2025.

Vishakh Padmakumar, Chen Yueh-Han, Jane Pan, Valerie Chen, and He He. Beyond memorization: Mapping the originality-quality frontier of language models. *arXiv e-prints*, pp. arXiv–2504, 2025.

Samuel J Paech. EQ-Bench creative writing benchmark v3. https://github.com/EQ-bench/creative-writing-bench, 2025.

Jonathan A Plucker, Ronald A Beghetto, and Gayle T Dow. Why isn't creativity more important to educational psychologists? potentials, pitfalls, and future directions in creativity research. *Educational psychologist*, 39(2):83–96, 2004.

Cheng Qian, Hyeonjeong Ha, Jiayu Liu, Bingxiang He, Jeonghwan Kim, Jiateng Liu, Bingxuan Li, Aditi Tiwari, Dwip Dalal, Zhenhailong Wang, et al. CreativityBench: Evaluating creative reasoning via affordance-based tool repurposing. In *Third Conference on Language Modeling*.

Yansheng Qiu, Haoquan Zhang, Zhaopan Xu, Ming Li, Diping Song, Zheng Wang, and Kaipeng Zhang. AI idea bench 2025: AI research idea generation benchmark. *arXiv preprint arXiv:2504.14191*, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026b. URL https://qwen.ai/blog?id=qwen3.8.

Kai Ruan, Xuan Wang, Jixiang Hong, Peng Wang, Yang Liu, and Hao Sun. Evaluating LLMs' divergent thinking capabilities for scientific idea generation with minimal context. *Nature communications*, 17(1):3625, 2026.

Mark A Runco and Garrett J Jaeger. The standard definition of creativity. *Creativity research journal*, 24(1):92–96, 2012.

Prabir Sarkar and Amaresh Chakrabarti. Assessing design creativity. *Design studies*, 32(4):348–383, 2011.

Jami J Shah, Steve M Smith, and Noe Vargas-Hernandez. Metrics for measuring ideation effectiveness. *Design studies*, 24(2):111–134, 2003.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. In *International Conference on Learning Representations*, volume 2025, pp. 94003–94092, 2025.

Mirac Suzgun, Luke Melas-Kyriazi, Suproteem Sarkar, Scott D Kominers, and Stuart Shieber. The harvard uspto patent dataset: A large-scale, well-structured, and multi-purpose corpus of patent applications. *Advances in neural information processing systems*, 36:57908–57946, 2023.

Apodex Team. Apodex-1.0: A verification-centric agent team for discoverative intelligence. Technical report, 2026a.

Gemma Team. Gemma 4 technical report, 2026b. URL https://arxiv.org/abs/2607.02770.

Yufei Tian, Abhilasha Ravichander, Lianhui Qin, Ronan Le Bras, Raja Marjieh, Nanyun Peng, Yejin Choi, Thomas L Griffiths, and Faeze Brahman. MacGyver: Are large language models creative problem solvers? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5303–5324, 2024.

Zihan Wang, Lam Nguyen, Zhengyang Zhao, Mengyue Yang, Chengwei Qin, Yujiu Yang, and Linyi Yang. CreativeBench: Benchmarking and enhancing machine creativity via self-evolving challenges. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 30904–30937, 2026.

Thomas B Ward, Steven M Smith, and Ronald A Finke. Creative cognition. *Handbook of creativity*, 189:212, 1999.

Junyi Ye, Jingyi Gu, Xinyun Zhao, Wenpeng Yin, and Guiling Wang. Assessing the creativity of llms in proposing novel solutions to mathematical problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 25687–25696, 2025.

Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. NoveltyBench: Evaluating language models for humanlike diversity. *arXiv preprint arXiv:2504.05228*, 2025.

A Appendix

A.1 Validation of Dimension-Separated Evaluation

We provide additional details on the controlled comparison between joint and dimension-separated evaluation. All response pairs, prompts, model configurations, and statistical procedures were frozen before evaluation.

A.1.1 Evaluation Protocols

Under *joint evaluation*, a judge assesses quality and novelty within the same call. Under *separate evaluation*, two independent calls assess quality and novelty, respectively. Both protocols compare two responses to the same task. The quality-only judge receives only quality criteria and admissible verification evidence, whereas the novelty-only judge additionally receives the frozen conventional-solution set $\mathcal{R}_x$.

Each response pair is evaluated in both A/B and B/A orders. After mapping the judgments back to response identities, a pair is considered *mirror-consistent* if the two orders yield the same winner or both yield a tie. Mirror consistency therefore measures robustness to response position, rather than agreement with a reference label.

Table 4: Mirror consistency of joint and dimension-separated evaluation across seven domains. Confidence intervals report the paired Separate−Joint difference.

Dimension	Joint	Separate	Difference	95% CI
Quality	61.14%	67.00%	**+5.86 pp**	$[+2.43, +9.29]$
Novelty	74.71%	77.71%	+3.00 pp	$[-0.71, +6.57]$

Table 5: Agreement with frozen CreativeMath committee labels under matched evaluation contexts.

Dimension	Joint	Separate	Difference	95% CI
Quality	71.02%	73.45%	**+2.43 pp**	$[+0.41, +4.57]$
Novelty	36.71%	38.46%	+1.75 pp	$[-4.04, +6.99]$

A.1.2 Cross-Domain Evaluation

We evaluate 700 frozen response pairs, with 100 pairs from each of the seven domains. Each pair contains two responses to the same task. During pair construction, responses are matched by quality and length to avoid trivial comparisons; model identities and existing evaluation scores are hidden from the judge. No responses are regenerated or selected using the experimental results.

The joint, quality-only, and novelty-only conditions use the same judge, decoding configuration, and output schema. We use deterministic decoding with thinking disabled and evaluate every pair in both response orders. Confidence intervals are obtained from 10,000 paired bootstrap samples over response pairs. McNemar's exact test is additionally used to compare the paired consistency outcomes.

Separating quality significantly reduces sensitivity to response order. Novelty exhibits a positive aggregate estimate, but its confidence interval includes zero. Domain-level results are heterogeneous: separation improves novelty judgments in some domains but not consistently across all seven domains. We therefore attribute the demonstrated reliability gain primarily to quality.

A.1.3 Matched-Context Validation on CreativeMath

We further conduct a matched-context evaluation using frozen CreativeMath (Ye et al., 2025) outputs and reference labels. The quality subset contains 226 response pairs from 95 underlying problems whose official correctness labels differ. The novelty subset contains 143 pairs from 66 problems for which both responses are correct but their official coarse-grained novelty labels differ.

Within each subset, the joint and separate conditions receive the same mathematical problem, reference solutions, response pair, and response order. The only experimental difference is whether the judge evaluates both dimensions or only the target dimension. All calls use Qwen3.8-27B with temperature zero, thinking disabled, and the same constrained output format. Every pair is evaluated in both A/B and B/A orders.

For each order, accuracy is computed against the frozen CreativeMath committee label. We report their average as

$$\mathrm{Acc}_{\mathrm{mean}} = \frac{1}{2}\left(\mathrm{Acc}_{A/B} + \mathrm{Acc}_{B/A}\right). \tag{13}$$

Because multiple response pairs may originate from the same mathematical problem, 95% confidence intervals are obtained from 10,000 bootstrap samples clustered by the underlying problem identifier.

The matched-context experiment confirms a statistically reliable improvement for quality. Novelty again favors separate evaluation in point estimate, but the interval includes zero. Its lower absolute accuracy reflects the deliberate selection of difficult pairs for which both responses are correct but receive different novelty labels.

A.1.4 SCOPE OF THE EVIDENCE

The results establish that dimension separation improves the positional robustness and reference agreement of quality judgments. They do not establish a statistically reliable aggregate improvement for novelty across all domains. Moreover, separate calls do not imply that the resulting quality and novelty scores are statistically independent, since the underlying response properties and judge preferences may remain correlated. We adopt separate evaluation primarily for demonstrated quality reliability, construct isolation, and auditability.

A.2 FORMULATION DETAILS

This section specifies the three creativity dimensions, the construction of their references, and the properties of the aggregation function.

A.2.1 QUALITY

For task $\tau_d = (x, \mathcal{C}_x, \mathcal{R}_x; \Omega_x)$, each $c \in \mathcal{C}_x$ denotes an explicit or implicit requirement governing the validity and effectiveness of a response. Let $\sigma_{d,c}(y) \in [0,1]$ measure the extent to which response y satisfies c. Quality is defined as

$$Q_d(y \mid \tau_d) = \sum_{c \in \mathcal{C}_x} \alpha_{d,c} \sigma_{d,c}(y), \qquad \alpha_{d,c} \geq 0, \quad \sum_{c \in \mathcal{C}_x} \alpha_{d,c} = 1. \tag{14}$$

The weights reflect the relative importance of the constraints. Whenever a task-native verifier is available, its result grounds the corresponding criterion; otherwise, satisfaction is assessed using the applicable domain-level criteria or instance-level rubric.

A.2.2 NOVELTY

Novelty is defined relative to the conventional solution families $\mathcal{R}_x$. Each entry represents a distinct strategy, mechanism, or structural pattern rather than a surface-level paraphrase. Every evaluated task contains at least one such family.

Let $\mathcal{A}_d$ denote the novelty dimensions for domain d, and let $S_{d,a}(y, r) \in [0,1]$ measure the similarity between response y and conventional family r along dimension a. We select the most similar families as

$$\mathcal{R}^{(K)}_{x,d,a}(y) = \operatorname*{TopK}_{r \in \mathcal{R}_x} S_{d,a}(y, r), \qquad K_x = \min(K, |\mathcal{R}_x|)\,, \tag{15}$$

and compute their mean similarity:

$$\overline{S}^{(K)}_{d,a}(y, \mathcal{R}_x) = \frac{1}{K_x} \sum_{r \in \mathcal{R}^{(K)}_{x,d,a}(y)} S_{d,a}(y, r). \tag{16}$$

Novelty is then

$$N_d(y \mid \tau_d) = \sum_{a \in \mathcal{A}_d} \gamma_{d,a} \left[1 - \overline{S}^{(K)}_{d,a}(y, \mathcal{R}_x)\right], \qquad \gamma_{d,a} \geq 0, \quad \sum_{a \in \mathcal{A}_d} \gamma_{d,a} = 1. \tag{17}$$

Using the Top-K mean reduces sensitivity to an anomalous reference or similarity estimate. The construction and freezing of $\mathcal{R}_x$ are detailed in Appendix A.10.5.

A.2.3 CREATIVE SPACE

Creative space is a property of the task rather than a generated response. Let $\mathcal{B}_d$ denote the relevant opportunity dimensions in domain d, including alternative valid mechanisms, representations, or objective gains. For each $b \in \mathcal{B}_d$, $\omega_{d,b}(\tau_d) \in [0,1]$ measures the opportunity afforded by the task along that dimension. We define

$$\Omega_d(\tau_d) = \sum_{b \in \mathcal{B}_d} \eta_{d,b} \omega_{d,b}(\tau_d), \qquad \eta_{d,b} \geq 0, \quad \sum_{b \in \mathcal{B}_d} \eta_{d,b} = 1. \tag{18}$$

The score is determined before response generation and remains fixed across models, responses, scoring modes, and evaluation methods. Thus, $\Omega_d(\tau_d)$ measures the opportunity for meaningful

creativity rather than whether a particular response exploits that opportunity. Its operationalization and validation are provided in Appendix A.14.1.

A.2.4 Reference Construction

The conventional solution set $\mathcal{R}_x$ is derived from source references, accepted solutions, and task metadata. Entries are normalized, deduplicated, grouped by substantive mechanism, and frozen before evaluation. Task constraints and reference families are never updated using evaluated model responses.

For deterministic quality estimation on tasks without native verifiers, a separate quality-anchor bank $\mathcal{Q}_x$ contains valid and complete responses. Unlike $\mathcal{R}_x$, which defines conventionality, $\mathcal{Q}_x$ provides positive evidence of requirement satisfaction. Its construction and validation are described in Appendix A.10.5.

A.2.5 Aggregation Properties

In the experiments, the domain-specific aggregation function is instantiated as a weighted geometric mean:

$$C_d(y \mid \tau_d) = Q_d(y \mid \tau_d)^{w_{d,Q}} N_d(y \mid \tau_d)^{w_{d,N}} \Omega_d(\tau_d)^{w_{d,\Omega}}, \tag{19}$$

where

$$w_{d,Q}, w_{d,N}, w_{d,\Omega} > 0, \qquad w_{d,Q} + w_{d,N} + w_{d,\Omega} = 1. \tag{20}$$

The function is non-decreasing in every dimension and satisfies

$$C_d(y \mid \tau_d) = 0 \quad \text{if} \quad Q_d(y \mid \tau_d) = 0,\ N_d(y \mid \tau_d) = 0,\ \text{or } \Omega_d(\tau_d) = 0. \tag{21}$$

The geometric mean therefore prevents a high value on one dimension from fully compensating for the absence of another. Because creative space is fixed within an instance, it does not alter within-instance model rankings; instead, it calibrates the relative value of instances in cross-instance aggregation. Scores reported on the $[0, 10]$ scale are obtained by multiplying the normalized scores by 10.

A.3 Details of Task Selection

Selection Principles. We select tasks that admit multiple substantively different yet valid responses, provide sufficient constraints or evidence for assessing quality, and support comparison with conventional solutions for assessing novelty. Each task is assigned to one primary domain according to its central creative operation, even when it involves knowledge from multiple domains. The first six domains are adapted from twelve public benchmarks; real-world application tasks are constructed separately from grounded application corpora.

Table 6 maps the heterogeneous benchmark criteria to the shared dimensions of quality, novelty, and creative space. These mappings provide common evaluation semantics rather than imply that every source benchmark explicitly reports all three dimensions.

Source Adaptation. We retain the task objectives, reference information, and evaluation evidence of each source benchmark while normalizing their interfaces. For benchmarks covering multiple domains, only the subsets matching the assigned domain are included. Table 7 summarizes the source-native criteria and representative task forms; detailed filtering, deduplication, and normalization procedures are provided in Appendix A.9.

Verification Coverage. Verifiability is an instance-level property rather than an intrinsic property of a domain. A task is judge-free when response validity can be determined through answer matching, program execution, deterministic constraint checking, or structured affordance matching. Hybrid tasks combine such signals with reference-grounded judgment, while open-ended tasks rely primarily on task-specific rubrics and supporting evidence. The symbols in Table 7 summarize the dominant verification forms within each domain and do not restrict the evaluation methods applied in CreativitySuite.

Table 6: Domain semantics of Quality, Novelty, and Creative Space. Superscripts indicate judge-free (ᵀ), hybrid (†), and LLM- or human-judged (‡) source protocols.

Domain	Quality	Novelty	Creative Space	Benchmark
Mathematical Reasoning	Correctness Rigor Requirement Coverage	Solution Paths Key Insights	Alternative Representations Alternative Proofs Insight Gains Generality Gains	CreativeMath‡ (Ye et al., 2025)
Code Generation	Correctness Robustness Implementation Quality	Algorithms Mechanisms Decomposition	Alternative Algorithms Alternative Designs Efficiency Gains Robustness Gains	CreativeBenchᵀ (Wang et al., 2026); NeoCoderᵀ (Lu et al., 2025)
Commonsense Reasoning	Semantic Correctness Clue Fit Explanatory Support	Lateral Connections Inference Paths	Alternative Connections Alternative Interpretations Explanatory Coverage Gains	CresOWLve† (Ismayilzada et al., 2026)
Physical Problem Solving	Physical Validity Affordance Fit Execution Safety	Object Repurposing Causal Paths	Alternative Affordances Alternative Mechanisms Effectiveness Gains Efficiency Gains	CreativityBenchᵀ (Qian et al.); MacGyver‡ (Tian et al., 2024)
Scientific Ideation	Scientific Validity Coherence Testability	Hypotheses Mechanisms Methods	Alternative Hypotheses Alternative Methods Information Gains Explanation Gains	LiveIdeaBench‡ (Ruan et al., 2026); ScienceAgentBench† (Chen et al., 2025b)
Creative Writing	Task Fulfillment Coherence Execution Quality	Content Expression Structure	Alternative Structures Alternative Expressions Emotional Gains Thematic Gains	CreativityPrism† (Hou et al., 2026b); CreataSetᵀ (Cao et al., 2026); LitBenchᵀ (Fein et al., 2026); CS4† (Atmakuru et al., 2024)
Real-World Applications	Goal Effectiveness Feasibility Completeness	Concepts Implementation Mechanisms	Alternative Product Mechanisms Alternative Service Mechanisms User Value Gains Practical Value Gains	CREATIVITYSUITE‡

A.4 CORPUS SOURCES AND CONSTRUCTION ROUTES

Source Corpora. We derive real-world application tasks from eight corpora containing products, services, technical artifacts, research outcomes, or communication materials. These corpora provide concrete needs, operating conditions, and external signals for constructing grounded tasks. External feedback is used only for sampling and screening, not as a direct creativity label. Table 8 summarizes the sources.

Construction Routes. After screening, records are transformed into self-contained tasks through two complementary routes. *Reconstruction* recovers the underlying problem from an existing outcome while concealing its identifiable solution. *Synthesis* uses a source record only as a contextual seed for a semantically distinct task. Table 9 summarizes the two routes.

[1] https://webrobots.io/kickstarter-datasets
[2] https://huggingface.co/datasets/lumalik/Product-Hunt-Data
[3] https://huggingface.co/datasets/ibragim-bad/github-repos-metadata-40M
[4] https://huggingface.co/datasets/jamescalam/ai-arxiv
[5] https://huggingface.co/datasets/AkashPS11/recipes_data_food.com
[6] https://huggingface.co/datasets/PeterBrendan/Ads_Creative_Text_Programmatic

Table 7: Task sources, source-native criteria, and representative examples across the seven domains.

Domain	Source	Source-Native Criteria	Representative Task	Verification
Mathematical Reasoning	CreativeMath (Ye et al., 2025)	Correctness and solution novelty	Produce a correct but non-standard solution to an olympiad problem given conventional solutions.	△
Code Generation	CreativeBench (Wang et al., 2026); NeoCoder (Lu et al., 2025)	Correctness, constraint satisfaction, and algorithmic novelty	Generate a program that passes the tests while avoiding specified solution techniques.	✓
Commonsense Reasoning	CresOWLve (Ismayilzada et al., 2026)	Answer correctness and lateral knowledge integration	Recover a missing concept by connecting distant factual and contextual clues.	△
Physical Problem Solving	CreativityBench (Qian et al.); MacGyver (Tian et al., 2024)	Affordance grounding, physical feasibility, and effectiveness	Repurpose an available object or one of its parts to achieve a constrained physical goal.	△
Scientific Ideation	LiveIdeaBench (Ruan et al., 2026); ScienceAgentBench (Chen et al., 2025b)	Originality, feasibility, clarity, and executable task success	Propose a scientific idea from minimal context or implement an executable analysis for a scientific task.	△
Creative Writing	CreativityPrism (Hou et al., 2026b); CreataSet (Cao et al., 2026); LitBench (Fein et al., 2026); CS4 (Atmakuru et al., 2024)	Quality, novelty, diversity, human preference, and constraint satisfaction	Write a coherent short story that incorporates unusual narrative constraints.	△
Real-World Applications	Ours	Novelty, feasibility, completeness, and practical value	Design a non-stigmatizing campaign that helps families access summer meal support under realistic communication constraints.	×

✓: primarily judge-free; △: mixed judge-free and judge-based task forms; ×: primarily judge-based.

Table 8: Corpus sources, grounding artifacts, selection signals, and representative tasks for real-world applications.

Source	Grounding Artifact	Selection Signal	Representative Task
Amazon Reviews 2023 (Hou et al., 2026a)	Product descriptions and functional attributes	Ratings and review counts	Design a reconfigurable kitchen–dining workstation for small spaces
Kickstarter [1]	Crowdfunding projects and real-world needs	Funding ratio, backer count, and platform recommendations	Design a low-cost emergency power supply for communities
Product Hunt [2]	Technology products and service concepts	Vote and discussion counts	Design a collaboration tool for independent developers
GitHub Repositories [3]	Open-source software and developer tools	Community interest and reuse	Design a data-synchronization system for low-connectivity environments
HUPD (Suzgun et al., 2023)	Patent abstracts, technical problems, and inventions	Patent examination status	Design a scalable and manufacturable flexible-energy device
arXiv AI/CS [4]	Research problems, methods, and technical systems	Research area and content completeness	Translate research principles into deployable applications
Food.com Recipes [5]	Dish compositions and cooking procedures	Ratings and review counts	Design a batch-preparable dish under dietary constraints
Programmatic Ads [6]	Advertising text and communication contexts	Content completeness and contextual specificity	Design a low-stigma communication campaign for public services

A.5 Fine-Grained Real-World Application Tasks

During data construction, we define a candidate taxonomy spanning consumer design, technical systems, research practice, and environmental settings. After corpus screening, quality review, and

Table 9: Construction routes for real-world application tasks.

Route	Task Construction	Evaluation Context
Reconstruction	Recover the context, requirements, constraints, and delivery objectives from a real-world outcome while concealing its identity and specific solution.	Original outcome, external feedback, and inferred conventional solutions.
Synthesis	Use a real-world record as a contextual seed to construct a semantically distinct task and its conventional solution families.	Conventional solutions and task-specific evaluation criteria.

cross-category deduplication, the frozen training and evaluation sets cover the 26 task types in Table 10.

Narrative writing, script writing, poetry, and world-building tasks are assigned to Creative Writing rather than Real-World Applications. Candidate categories without sufficient high-quality data are excluded. The taxonomy controls scenario coverage and sampling diversity; it is not an additional creativity dimension.

Table 10: Fine-grained taxonomy and representative examples of real-world application tasks.

Task Type	Representative Example
Consumer and Design	
Product and Industrial Design	Design a reconfigurable kitchen workstation for food preparation, dining, and storage in small households.
Advertising	Design a clear, credible, and non-stigmatizing campaign for a summer meal assistance program.
Branding and Naming	Develop a name, brand positioning, and extensible visual identity for a low-sugar beverage.
UX and Service Design	Redesign an outpatient appointment process to reduce waiting times and missed appointments.
Game and Activity Design	Design an outdoor activity for young children that combines cooling, physical exercise, and nature learning.
Culinary Innovation	Design a nutritious meal subject to allergen, cost, and batch-preparation constraints.
Visual and Graphic Design	Design an emergency-evacuation graphic that can be quickly understood in a complex public building.
Content and Short-Form Video	Plan a short-video series that promotes household waste sorting and sustained participation.
Business Model and Entrepreneurship	Develop a sustainable service model combining the repair, sharing, and recycling of unused appliances.
Educational Activity Design	Design a repeatable climate-science classroom activity using commonly available materials.
Fashion and Apparel Design	Design a modular commuter jacket adaptable to different weather conditions and use cases.
Event and Community Planning	Transform an underused public space into a recurring community-led activity.
Music and Sound Design	Design a recognizable and low-noise acoustic alert that improves the safety of low-speed electric vehicles.
Technology and Systems	
Software and System Design	Design a collaborative editing system that operates under intermittent, low-bandwidth connectivity.
Algorithm and Data Structure Design	Design an efficient and stable delivery-scheduling algorithm under dynamic constraints.
AI Application and Agent Design	Design a customer-service agent with evidence verification, human review, and failure recovery.
Data and Infrastructure Design	Design a data-collection pipeline tolerant of disconnections and delays across distributed sensors.

Continued on next page

Table 10 continued

Task Type	Representative Example
Human–Computer Interaction	Design a low-distraction interface for factory operators who cannot continuously monitor a screen.
Hardware and IoT Design	Design a low-power, easy-to-install household leak-detection system with a low false-alarm rate.
Engineering and Mechanical Design	Design a manufacturable portable passive-cooling device that requires no continuous power supply.
Security and Privacy Design	Design an authentication mechanism that balances convenient access with privacy protection in shared workspaces.
Data Visualization	Design an interface that jointly communicates air-quality trends, spatial variation, and predictive uncertainty.
Research and Interdisciplinary Applications	
Research and Experimental Design	Design an experiment comparing biodegradable packaging materials under realistic environmental conditions.
Interdisciplinary Innovation	Combine origami structures and soft-robotics principles to design an affordable rehabilitation device.
Environment and Space	
Architectural, Urban, and Spatial Design	Redesign a heat-exposed neighborhood to provide shade, ventilation, drainage, and space for public activities.
Sustainability and Environmental Design	Develop a circular food-waste system connecting restaurants, residents, and processing facilities.

A.6 Corpus Screening Details

Real-world corpora undergo rule-based filtering, stratified organization, semantic selection, and quality screening. Every stage writes its output to a versioned candidate pool with source identifiers, enabling resumable processing and provenance auditing.

Rule-Based Filtering. Rule-based filtering removes records that cannot support task construction, including missing fields, insufficient text, malformed records, invalid status values, and content without substantive information. Because corpus structures and feedback signals differ, we apply the source-specific criteria in Table 11. Smaller corpora and sources without reliable external feedback receive only basic completeness checks to avoid excessive filtering.

Stratified Organization. We organize candidates by source category and external feedback without treating feedback strength as creativity. Amazon and Food.com use combinations of ratings and review counts; Kickstarter uses funding ratio; Product Hunt and GitHub use votes and community interest, respectively; and HUPD uses examination status. Because arXiv and Programmatic Ads lack reliable feedback signals, each is assigned a single signal stratum.

Continuous signals are divided into high, medium, and low strata, while HUPD is stratified by accepted and rejected status. All strata are retained to prevent the candidate pool from containing only highly popular outcomes. Sources with explicit categories are additionally stratified by product category, crowdfunding category, CPC patent class, recipe category, or research field. No fixed quota is imposed at this stage, except that Amazon uses the additional processing thresholds described above because of its scale.

Semantic Selection. We encode each record using Qwen3-Embedding-8B, based on its title, category, and main text, and cluster records within each source. Near-duplicates are removed within clusters, prioritizing records with stronger external signals or more complete information. Candidates are then interleaved across signal strata, source categories, and semantic clusters to preserve topical and signal coverage.

The encoder input contains the title, category, and at most 600 characters of the main text. Embeddings are normalized before cosine-based clustering. For a source with n candidates, the number of clusters is

$$k_{\text{cluster}} = \min\big(512, \max\big(8, \lfloor 2\sqrt{n} \rfloor\big)\big)\,. \tag{22}$$

Table 11: Primary rule-based filtering criteria for each source corpus.

Source	Filtering Criteria
Amazon Reviews	Require a product description of at least 200 characters, a rating of at least 3.5, and at least 50 reviews. Remove replacement parts, refills, and compatibility-only accessories. Because the eligible pool remains large, the subset entering semantic selection additionally requires a rating of at least 4.2 and at least 600 reviews.
Kickstarter	Retain only completed projects marked successful or failed, with descriptions of at least 80 characters and funding goals of at least $1,000.
Product Hunt	Require a complete product description and at least 10 votes. Remove records without a substantive product or service concept.
GitHub	Require an English repository description of at least 40 characters and at least 50 stars. Remove resource lists, tutorial collections, and repositories without independent functionality.
HUPD	Require a patent abstract of at least 200 characters and an examination status of accepted or rejected. Remove pending applications and records without a substantive technical description.
arXiv AI/CS	Require an abstract of at least 100 characters containing an identifiable research problem, method, or technical system.
Food.com	Require a rating of at least 3.5, at least two reviews, and a dish description of at least 40 characters.
Programmatic Ads	Require advertising text of at least 30 characters. Remove corrupted text, incomplete templates, and records without identifiable communicative content.

Clustering uses 15 iterations and random seed 42. Records with cosine similarity above 0.92 within a cluster are treated as near-duplicates.

Quality Screening. Quality screening comprises two complementary assessments. First, Qwen3-Next-80B scores the *information sufficiency* of each record on a 0–3 scale:

- 0: insufficient substantive information for task construction;
- 1: limited information supporting only a generic task;
- 2: sufficient information for a self-contained task with context and constraints;
- 3: rich information supporting a distinctive real-world task.

For large corpora, records scoring at least 2 enter the high-quality candidate pool, while records scoring 1 are retained as supplementary training candidates. For small corpora, records scoring at least 1 are retained. The prompt includes at most 900 characters. Decoding uses temperature 0.1 and a maximum of 200 tokens. Two Qwen3-Next-80B inference replicas score the formal pool in parallel. Unparseable outputs are not automatically promoted to the high-quality pool.

Second, Qwen3-235B-A22B assesses the *creativity suitability* of high-quality candidates on another 0–3 scale:

- 0: near-unique solution, specification selection, or routine procedure;
- 1: limited opportunity for local improvement;
- 2: multiple valid solutions and a clear design space;
- 3: multiple substantively different solution paths and substantial creative space.

Large corpora require a creativity-suitability score of at least 2, whereas small corpora require at least 1. Decoding again uses temperature 0.1 and a maximum of 200 tokens. Information sufficiency measures whether a complete task can be constructed; creativity suitability measures whether that task can meaningfully evaluate creativity. Neither criterion substitutes for the other.

Retention Statistics. Table 12 reports retention after each stage. Rule-based filtering yields 3,763,803 eligible records. Amazon contributes 3,437,116 records and therefore receives an additional processing threshold. A total of 829,958 records enter semantic selection, of which 515,198

Table 12: Retention across screening stages. Parentheses report retention relative to the preceding stage.

Source	Semantic Input	After Semantic Selection	Information-Sufficient	Creativity-Suitable
Amazon	503,271	209,235 (41.6%)	207,707 (99.3%)	189,037 (91.0%)
Kickstarter	129,058	127,309 (98.6%)	115,195 (90.5%)	114,628 (99.5%)
Product Hunt	21,051	17,571 (83.5%)	17,540 (99.8%)	17,539 (100.0%)
GitHub	7,907	7,878 (99.6%)	7,702 (97.8%)	7,307 (94.9%)
HUPD	166,574	151,162 (90.7%)	150,525 (99.6%)	119,125 (79.1%)
arXiv	423	422 (99.8%)	422 (100.0%)	422 (100.0%)
Food.com	713	707 (99.2%)	700 (99.0%)	700 (100.0%)
Programmatic Ads	961	914 (95.1%)	821 (89.8%)	821 (100.0%)
Total	**829,958**	**515,198 (62.1%)**	**500,612 (97.2%)**	**449,579 (89.8%)**

Reconstruction Prompt

The following record describes an existing real-world outcome. Infer the open-ended task that this outcome was designed to address so that another model can independently propose a solution.
Describe only the need, problem, target setting, and constraints. Do not reveal the outcome's name, identifying information, specific form, technical route, or core implementation. The task must be self-contained, grounded in a realistic setting, and admit multiple substantively different valid solutions.
Task type: {`task_type`}
Real-world artifact: {`artifact`}
Return only the following JSON:

```
{
  "title": "...",
  "scenario_context": "...",
  "challenge": "...",
  "requirements": ["...", "..."],
  "deliverable": "...",
  "conventional_solutions": ["...", "..."],
  "evaluation_focus": "..."
}
```

Figure 5: Reconstruction prompt for recovering the underlying task of a real-world artifact.

remain after semantic filtering. Information screening identifies 500,612 high-quality candidates, and creativity-suitability screening retains 449,579 records for formal task construction.

Beyond the 500,612 high-quality candidates, information-sufficiency screening retains 13,144 borderline records as supplementary training candidates. An additional 1,442 low-quality records are retained only for auditing. Candidate pools use fixed ordering, source-aware deduplication, and content fingerprints. Subsequent construction consumes eligible records sequentially, ensuring consistent reruns and incremental extensions.

A.7 Details of Real-World Task Construction

Structured Representation. Both construction routes use the intermediate representation

$$z = (t, b, h, \mathcal{C}, o, \mathcal{R}, e)\,, \tag{23}$$

where t is the title, b the scenario context, h the creative challenge, $\mathcal{C}$ the key constraints, o the required deliverable, $\mathcal{R}$ the conventional solution families, and e the evaluation focus. The public fields $(t, b, h, \mathcal{C}, o)$ are rendered deterministically into the model-visible task. The conventional solutions $\mathcal{R}$ and evaluation focus e remain hidden as evaluation context.

Reconstruction Prompt. For a real-world artifact a, we recover its underlying problem using the prompt in Figure 5.

Synthesis Prompt

Design a meaningful, open-ended creative task grounded in a real-world setting.
The task must address a specific need, problem, or usage context. It must be self-contained and include sufficient background and actionable constraints. It must not have a unique solution or reduce to a generic aspiration. The record below is contextual inspiration only; do not reproduce its problem wording, product form, or core solution.
Task type: {task_type}
Contextual seed: {seed_artifact}
Return only the same JSON fields used for Reconstruction.

Figure 6: Synthesis prompt for constructing a semantically distinct task from a corpus seed.

The input artifact consists of its name, category, and main description, truncated to 1,600 characters. The artifact and its external feedback are not included in the public task and are retained only as internal grounding evidence.

Synthesis Prompt. For a corpus seed a, we construct a semantically distinct task using the prompt in Figure 6.

The seed contains the artifact name, category, and at most 220 characters of content. Its source, identifier, and link are retained for provenance but are not exposed to evaluated models.

Deterministic Rendering. Structured records are rendered in the order

$$t \to b \to h \to \mathcal{C} \to o. \tag{24}$$

The resulting task contains a title, scenario context, creative challenge, key constraints, and required deliverable. Models are asked to provide a complete and concrete proposal with explanations of its novelty and feasibility. Conventional solutions and evaluation criteria are excluded from the public prompt to avoid revealing reference information.

Structural Validation. A generated record must satisfy all of the following conditions:

- the scenario context and creative challenge are non-empty;
- the scenario context contains at least 30 characters;
- at least two key constraints and two conventional solutions are provided;
- the rendered task contains at least 150 characters;
- the output conforms to the required JSON schema;
- the public task does not disclose identifying information or the core solution of the source artifact.

Invalid generations are retried automatically. After repeated failure, the record is rejected and replaced by the next eligible item in the candidate pool.

Generation Configuration. Formal task construction uses GPT-5.5 with a maximum output length of 2,600 tokens. Reconstruction uses temperature 0.7 to prioritize faithful recovery, whereas Synthesis uses temperature 0.9 to encourage task diversity. Reconstruction permits at most two attempts and Synthesis at most three. Each record stores the generation model, construction route, source identifier, decoding configuration, and processing status for reproducibility and audit.

A.8 Reference-Response Construction

Candidate Generation. For each training task, four heterogeneous teacher models independently generate candidate responses. Teachers receive only the public task and cannot access the original artifact, external feedback, conventional solutions, or hidden evaluation context. This prevents direct copying of source outcomes into the supervision targets.

Generation Prompt

Propose an original solution to the following creative task. Briefly consider several possible directions, then fully develop the selected proposal, including its central insight, operating mechanism, implementation details, and expected effects. The proposal must be concrete, feasible, and satisfy all task constraints. Avoid generic statements, conversational preambles, and references to the prompt itself.
Task: {`question`}

Figure 7: Generation prompt used by each teacher model for candidate generation.

Evaluation Prompt

You are a strict reviewer of creative proposals. Evaluate anonymous candidates A–D against the task and each rubric item. Every score must be an integer not exceeding the corresponding maximum. Judge only the proposal content, constraint satisfaction, and feasibility; do not reward length or writing style. Return only structured scores in rubric order.

Figure 8: Summary of the evaluation prompt used for rubric-based selection.

Each candidate must present a complete proposal covering its central concept, implementation mechanism, constraint handling, and expected effects. The generation prompt is shown in Figure 7.

To reduce teacher-specific stylistic bias, models sample independently and do not observe one another's outputs. Model identifiers, decoding parameters, and prompt versions are retained with each generation.

Rubric-Based Selection. An independent evaluator performs listwise assessment of the candidate responses. For each task, the four responses are deterministically shuffled using the task identifier and anonymized as A–D, reducing model-identity and position bias. The evaluator assigns an integer score to every active instance-level rubric item and does not directly produce an overall score.

Let $s_{i,m,j}$ and $M_{i,j}$ denote the awarded and maximum scores for criterion j of candidate m on task i. The normalized candidate score is

$$R_{i,m} = \frac{\sum_{j=1}^{J_i} s_{i,m,j}}{\sum_{j=1}^{J_i} M_{i,j}}. \tag{25}$$

The item maxima encode their within-task weights. We select the candidate corresponding to

$$\arg\max_{m} R_{i,m} \tag{26}$$

as the supervision target. Ties are explicitly recorded and resolved, when a single target is required, using a fixed versioned order. Creative Space is a task-level attribute and does not affect candidate ranking.

The evaluation prompt is summarized in Figure 8.

Validity Filtering. Candidate and evaluation completeness are checked before selection. Failed, empty, excessively short, prompt-referencing, or incomplete responses are excluded. A task is not used for supervision if any candidate, instance-level rubric, or valid criterion score is missing.

Evaluation outputs must cover every active rubric item, and each score must lie within its permitted range. Parsing failures receive a limited number of retries; unrecoverable records are rejected. The selected response must also pass task-constraint, format-completeness, and meta-narrative checks. Training and evaluation data are separated by task identifier, and evaluation tasks and all associated candidate responses are excluded from training.

Provenance and Copyright. Each supervision record retains its task identifier, corpus source, construction route, teacher model, prompt version, criterion-level scores, and selection result. Third-party artifacts are used only for task construction and internal grounding and are never copied di-

rectly as training responses. Teacher models generate supervision targets solely from the derived public tasks.

For sources with unclear or restrictive redistribution terms, released data exclude the original artifact text and contain only distributable derived fields, source identifiers, and reconstruction procedures. All remaining content follows the applicable source licenses. The complete license inventory and release scope are documented in the dataset statement.

A.9 Details of Task Curation

This section describes the curation of evaluation data, supplementary training data, protocol adaptation, rejection sampling, and construction audits.

A.9.1 Evaluation-Data Curation

Source preservation and structural validation. We retain the task semantics, reference information, and native verification assets of each source benchmark. Samples are excluded when they contain missing problem statements or required references, invalid labels, incomplete tests, unrecoverable answers, or incompatible evaluation configurations. For executable tasks, we additionally verify that the provided interfaces, tests, and expected outputs are internally consistent.

Cross-source deduplication. When a task appears in both its original benchmark and an aggregated dataset, we retain only the original source. For example, mathematical tasks originating from CreativeMath (Ye et al., 2025) are not duplicated through CreativityPrism (Hou et al., 2026b). We first detect exact matches after removing fixed instructions and output wrappers, and then retrieve semantic near-duplicates. Candidate matches are manually examined for equivalent problem structures, answer spaces, and solution mechanisms. Only one version of an equivalent task is retained, and duplicates across training, validation, and test splits are removed to prevent leakage.

Quality auditing. Quality auditing combines source-native validators, model-based review, and human adjudication. It targets incorrect answers, defective prompts, incomplete tests, ambiguous labels, and conflicts between task semantics and evaluation rules. Repairs are restricted to objectively recoverable answers, labels, tests, or verifier configurations; repaired samples must pass the full validation pipeline again. Samples requiring subjective reconstruction are discarded.

Verification harmonization. Native objective evidence is preserved whenever available, including answer matching, program execution, formal constraint checking, and structured affordance matching. Tasks without sufficient deterministic evidence use reference-grounded model evaluation. CresOWLve (Ismayilzada et al., 2026) supports both routes: each task maintains accepted- and rejected-answer pools; an answer appearing in neither pool is adjudicated by an LLM judge and then added to the corresponding pool. This produces the judge-free and model-based verification routes summarized in Figure 4.

A.9.2 Supplementary Training-Data Generation

Source prioritization. We prioritize the official training splits and complete solutions released by the source benchmarks. Supplementary samples are constructed only when a domain lacks training data or when its existing data do not cover major task forms in the evaluation set. Records duplicated through aggregated benchmarks are not reused.

Construction protocol. Each supplementary sample is produced in two independent stages. A task constructor receives the target domain, task form, capability requirements, and output protocol, and generates a public problem, explicit constraints, and private verification evidence. A separate solver receives only the public problem and cannot access the source seed, reference solution, or private verification information. This separation prevents construction-time answer cues from entering the supervision target. Table 13 summarizes the domain-specific requirements.

Task-construction prompt. All domains share the core construction contract in Figure 9, specialized through domain and task-type fields.

Table 13: Construction and validation of supplementary training data.

Domain	Task Construction	Supervision Target	Primary Evidence
Mathematical Reasoning	Construct new problems with different contexts, numerical structures, or proof strategies.	Complete solutions or proofs.	Answer recomputation and proof review.
Code Generation	Generate function specifications, public examples, private oracles, and tests.	Executable programs.	Syntax checks, sandboxed execution, and hidden tests.
Commonsense Reasoning	Construct questions with sufficient clues and a unique semantic answer.	Answers with supporting reasoning.	Independent solving and answer-equivalence checks.
Physical Problem Solving	Vary goals, available objects, contact relations, and physical mechanisms.	Objects, components, and implementation procedures.	Geometry, force, stability, material, and safety checks.
Scientific Ideation	Construct new research questions, hypotheses, or executable research tasks.	Research proposals or executable artifacts.	Scientific validity, testability, or execution checks.
Creative Writing	Construct open generation, explicit-constraint, or preference tasks.	Written works, constraint-following responses, or choices.	Constraint checks, independent evaluation, or source labels.

Task-Construction Prompt

Given the domain, task type, target capability, and output requirements, construct a new self-contained task. Preserve the target capability and approximate difficulty, but change the concrete context, entity relations, and solution mechanism. Do not reproduce the source problem or reveal its answer. Return the public problem, task constraints, response contract, and any answer, tests, or evaluation evidence used only for verification.

Figure 9: Core task-construction contract shared by all domains.

Code constructors additionally produce a private oracle and candidate tests, whose expected outputs are obtained by executing the oracle. Mathematics, commonsense, and physical tasks require independently checkable answers or mechanistic evidence. Open-ended science and writing tasks instead include task-specific criteria without assuming a unique answer.

Independent-solving prompt. Supervised responses are generated independently from the public task using the prompt in Figure 10.

For code tasks, the solver sees only the specification and public examples. For open-ended tasks, it sees only the public context, constraints, and required deliverable. Private verification evidence is never included in the solver input.

A.9.3 Protocol Adaptation

Canonical representation. Source benchmarks differ substantially in prompt organization, answer serialization, and parsing rules. We therefore convert each sample into a protocol-independent semantic record containing its public problem, constraints, semantic answer, verification assets, task type, and provenance. This representation separates task semantics from source-specific wrappers.

Unified training protocol. All samples use a common conversational interface while retaining task-dependent response contracts, as summarized in Table 14. Intrinsic requirements such as function signatures, answer options, proof obligations, file artifacts, and writing constraints are preserved. Parser-specific XML tags, rigid JSON wrappers, and benchmark-specific entry points are

Independent-Solving Prompt

Complete the following task independently and follow all constraints and output requirements. The response must be complete, specific, and verifiable. Do not refer to the data-construction process, source samples, hidden references, or evaluation instructions.
Task: {public_problem}

Figure 10: Independent-solving prompt used to generate supervised responses.

Table 14: Task forms and response contracts used in the unified training protocol.

Domain	Task Form	Response Contract
Mathematical Reasoning	Short answer / proof	A complete derivation with an explicit final answer, or a complete proof with a clear conclusion.
Code Generation	Callable / standard input–output	An executable function with the required signature, or a complete standard input–output program.
Commonsense Reasoning	Answer / explanation	A concise natural-language answer, with an explanation only when requested.
Physical Problem Solving	Affordance / procedure	The object, component, and operating mechanism, or a physical solution with the necessary components and steps.
Scientific Ideation	Idea / execution	A complete research proposal, or an executable workflow containing the required code and artifacts.
Creative Writing	Generation / choice	A work satisfying the specified genre and constraints, or a choice label in the required format.
Real-World Applications	Solution	A complete and feasible proposal satisfying the stated needs and implementation constraints.

removed unless they are part of the task itself. Thus, adaptation standardizes the training interface rather than the semantic form of the answer.

Representative adaptations. Table 15 illustrates representative transformations. Adaptation changes only the prompt wrapper and answer serialization; task semantics, supervision targets, and verification conditions remain unchanged.

Each task has only one training representation, preventing the same semantic sample from being counted repeatedly under different protocols. Complete field mappings, response contracts, and conversion records are retained with each data version.

A.9.4 Rejection Sampling

Supplementary samples undergo four classes of checks: structural validity, correctness or feasibility, substantive novelty, and data isolation. A sample enters the final training set only after passing every applicable check.

Structural validation. The public problem, constraints, supervision target, verification assets, and response contract must be complete and mutually consistent. Code tasks require a valid interface and executable program; mathematical tasks require an unambiguous problem and answer objective; physical tasks must specify the goal and available resources; and science and writing tasks require an explicit deliverable. Empty responses, placeholders, truncated generations, invalid labels, and protocol mismatches are rejected.

Table 15: Representative adaptations from source-specific interfaces to the unified training protocol.

Task	Source Interface	Unified Training Interface
Mathematical option	The source parser extracts an answer from free-form text.	Retain the full derivation and explicitly state the option label and semantic answer.
Code callable	Benchmark-specific code wrapper.	Return an executable Python function with the required signature.
Code input–output	Benchmark-specific program entry point.	Return a complete standard input–output program.
Commonsense answer	Answer label or source-specific marker.	Return a concise natural-language answer while permitting equivalent aliases.
Physical affordance	Rigid fields or a JSON object.	Describe the object, component, usage procedure, and physical mechanism in natural language.
Writing preference	Source-specific option representation.	Return `Answer: A` or `Answer: B`.

Review Prompt

Determine whether the candidate is suitable as a training sample. Accept it only if the task is self-contained, the supervision target is correct or feasible, all essential constraints are satisfied, and the candidate differs substantively from the source seed and existing data. A rejection must identify a specific fatal defect, such as an incorrect answer, non-executable program, invalid proof, physical contradiction, missing constraint, or near-reproduction of the source task. Do not reject a candidate solely for style, length, or optional improvements.

Figure 11: Core review contract used in independent review.

Correctness and feasibility. Deterministic evidence is used whenever possible. Mathematical short-answer and commonsense tasks undergo answer verification. Code is subjected to syntax checks, public tests, private tests, and boundary cases. Explicit writing constraints are checked individually. Mathematical proofs are independently re-derived or challenged with counterexamples. Physical proposals are examined for object availability, contact relations, forces, stability, material properties, and safety. Scientific ideas are assessed for consistency between the research question and mechanism and for testability; executable scientific tasks additionally undergo interface checks or actual execution.

Novelty and contamination control. Candidates are first compared with their source seeds, the existing training pool, and the complete evaluation set through normalized exact matching. Candidates passing this stage are encoded with Qwen3-Embedding-8B and retrieved against their nearest semantic neighbors. Samples exceeding the threshold for the corresponding construction route are manually reviewed for shared problem structures and core solution mechanisms rather than surface wording.

The final training and evaluation sets have zero normalized exact overlap. A historical audit identified approximately 18 scientific training tasks with potential leakage; these tasks were quarantined from every training view, without modifying the evaluation set.

Independent review. Later large-scale batches are primarily generated with Claude Opus 4.8 and independently reviewed by GPT-5.5 for correctness, training value, and substantive novelty. Semantic deduplication is performed locally with Qwen3-Embedding-8B. Code and selected scientific execution tasks use separate architect and solver models, whose exact versions are frozen in the batch manifests.

A model-reported error sends the sample to a review queue rather than causing automatic rejection. Disputed mathematical proofs and physical proposals receive additional counterexample search, deterministic computation, or human review. The core review contract is shown in Figure 11.

Table 16: Representative decisions in supplementary training-data curation.

Task Type	Candidate	Evidence	Decision
Mathematical proof	A proof using a construction distinct from the source.	Independent derivation succeeds, no counterexample is found, and the proof strategy differs from the seed.	Accept
Code generation	A program that passes public examples but fails on a boundary input.	Its output disagrees with the private oracle and hidden tests.	Reject
Physical problem solving	A new combination of objects that assumes a nonexistent anchor point.	Contact and stability analysis shows that the proposal cannot be executed.	Reject
Scientific ideation	A testable new hypothesis with a controlled experiment.	The question and mechanism are consistent, the variables and validation path are explicit, and the proposal does not reproduce the seed.	Accept

Rejection criteria. A candidate is rejected if any of the following conditions holds:

- the problem, answer, constraints, or verification assets are missing;
- answer verification, program execution, or a source-native validator fails;
- a proof contains a logical gap or counterexample, or a physical proposal is infeasible;
- an open-ended task lacks a clear objective, valid deliverable, or basic scientific plausibility;
- the candidate differs from its seed only through entity substitution or surface paraphrasing;
- an unresolved exact or semantic near-duplicate exists in the training or evaluation pool; or
- the response exposes construction prompts, hidden references, or other private information.

Objectively repairable terminology, notation, labels, or verifier configurations may be corrected once. The repaired sample must then pass all applicable checks again; otherwise, it is excluded.

A.9.5 Implementation and Audit

Each construction batch records the task seed, full prompts, model identifiers, decoding configurations, raw generations, validation results, semantic neighbors, repairs, and final decisions. Only samples passing the complete pipeline enter the released data; rejected candidates and intermediate versions remain in a separate audit directory.

For example, a large-scale seed-transfer batch for mathematics and physical problem solving began with 1,200 training seeds and produced 1,096 structurally valid candidates. Independent review accepted 512, semantic deduplication retained 511, and final critical-error auditing and repair yielded 499 training samples. Thus, successful generation alone does not imply admission. Candidate counts, stage-wise retention rates, rejection reasons, and model configurations for other batches are recorded in the released construction manifests.

A.9.6 Representative Cases

Table 16 summarizes representative decisions. The examples illustrate the review logic; complete records are retained in the audit materials.

A.10 Evaluation Protocol Details

This section specifies the shared scoring contract, verifier integration, creative-space estimation, reference-bank construction, evaluator procedures, prompt contracts, and reproducibility controls.

Table 17: Principal verification routes across the seven domains.

Domain	Verified Component	Primary Evidence
Mathematical Reasoning	Answer correctness or proof validity	Answer matching, symbolic recomputation, or proof verification
Code Generation	Functional correctness	Sandboxed execution and frozen tests
Commonsense Reasoning	Semantic correctness	Reference answers, accepted aliases, and cached semantic adjudication
Physical Problem Solving	Affordance and physical validity	Object–part consistency and feasibility checks
Scientific Ideation	Scientific validity or execution correctness	Structured validity checks or task-native execution
Creative Writing	Constraint or preference correctness	Constraint checks or reference preference labels
Real-World Applications	Requirement satisfaction and feasibility	Structured constraints and task-specific feasibility evidence

A.10.1 Shared Evaluation Contract

All methods evaluate the same stored response and report Quality, Novelty, Creative Space, and overall creativity on a 0–10 scale. Scores may be multiplied by ten for presentation on a 0–100 scale; no run-dependent normalization is applied.

Quality and Novelty are evaluated independently. A Quality call receives only quality criteria and applicable verifier evidence, whereas a Novelty call receives only novelty criteria and conventional-solution references. Neither call observes the score or rationale produced for the other dimension. Creative Space is computed separately from the task and is not judged from the candidate response.

Scoring modes. For dimension $k \in \{Q, N\}$, the domain-level mode averages the corresponding domain criteria:

$$E_{i,k}^{\text{domain}} = \frac{1}{|\mathcal{U}_{d_i,k}|} \sum_{u \in \mathcal{U}_{d_i,k}} e_{i,k,u}, \tag{27}$$

where $\mathcal{U}_{d_i,k}$ contains the subdimensions defined in Table 6. The instance-level mode normalizes the task-specific rubric scores:

$$E_{i,k}^{\text{instance}} = 10 \frac{\sum_{j \in \mathcal{J}_{i,k}} r_{i,k,j}}{\sum_{j \in \mathcal{J}_{i,k}} r_{i,k,j}^{\max}}. \tag{28}$$

The maximum score of each rubric item determines its relative weight. Domain-level and instance-level scores are reported separately.

A.10.2 Verifier Integration

A task-native verifier returns $v_i \in [0, 1]$. Table 17 summarizes the principal routes; verifiability is determined at the instance level rather than solely by domain.

For model-based evaluation, objective correctness contributes 0.60 of Quality, while the remaining quality criteria jointly contribute 0.40:

$$E_{i,Q}^{(m)} = 0.6(10v_i) + 0.4E_{i,Q}^{(m,\text{rem})}. \tag{29}$$

The verifier replaces only the corresponding correctness component; it does not affect Novelty or Creative Space. Without an applicable verifier, Quality is computed entirely from the active criteria. Infrastructure failures are recorded as unavailable evidence rather than failed verification.

For commonsense tasks, a response is first matched against frozen reference answers and aliases. Previously adjudicated correct and incorrect responses are retrieved from a question-specific cache. Only unseen responses are submitted for semantic adjudication, after which the corresponding cache is updated. Initial and final cache hashes, updates, and conflicting verdicts are retained for reproducibility.

Table 18: Frozen parent-dimension weights for creativity aggregation.

Domain	$w_{d,Q}$	$w_{d,N}$	$w_{d,\Omega}$
Mathematical Reasoning	0.50	0.40	0.10
Code Generation	0.50	0.40	0.10
Commonsense Reasoning	0.50	0.40	0.10
Physical Problem Solving	0.50	0.40	0.10
Scientific Ideation	0.45	0.40	0.15
Creative Writing	0.45	0.40	0.15
Real-World Applications	0.40	0.40	0.20

A.10.3 Creative-Space Estimation

Creative Space is computed once per task, before candidate generation. Let $\mathcal{R}_{x_i}$ contain at most six mechanistically distinct conventional solution families and let $\widetilde{\mathcal{T}}_{x_i}$ contain at most six task-specific creative templates after semantic deduplication. We define

$$n_i = \sum_{r \in \mathcal{R}_{x_i}} \sum_{t \in \widetilde{\mathcal{T}}_{x_i}} \mathbb{I}_i(r, t), \tag{30}$$

where $\mathbb{I}_i(r, t) = 1$ only when applying template t to family r produces a concrete, valid, and substantively distinct extension. The score is

$$\Omega_i = 10 \frac{\log(n_i + 2)}{\log 38}. \tag{31}$$

The logarithm captures diminishing marginal information from additional paths; the denominator follows from the maximum of $6 \times 6 = 36$ valid pairs.

The construction model proposes templates and, for every accepted family–template pair, provides its adaptation, incremental benefit, and task-fit justification. Deterministic procedures validate identifiers, remove duplicate templates and pairs, count valid applications, and apply the frozen normalization. Wording variants and merely conceivable ideas are not counted. Missing or unaudited artifacts remain unavailable rather than receiving a default score. Candidate responses and response-specific rubrics are excluded from this process.

A.10.4 Aggregation

Overall creativity is recomputed locally:

$$C_i^{(m)} = 10 \left(\frac{E_{i,Q}^{(m)}}{10} \right)^{w_{d_i,Q}} \left(\frac{E_{i,N}^{(m)}}{10} \right)^{w_{d_i,N}} \left(\frac{\Omega_i}{10} \right)^{w_{d_i,\Omega}}. \tag{32}$$

All weights are positive and sum to one. Any overall score generated directly by an evaluator is discarded.

Quality receives greater weight in domains with stronger validity requirements, whereas Creative Space receives greater weight in more open-ended domains. Scores are first aggregated within each domain and then macro-averaged across the seven domains.

A.10.5 Reference Banks

Conventional-solution bank. Each entry in $\mathcal{R}_{x_i}$ represents a distinct conventional solution family through its central strategy, mechanism, and representative realization. The bank is initialized from source references, available solutions, and task metadata. When these sources provide insufficient coverage, additional candidates are elicited using the prompt in Figure 12.

Formatting artifacts and task-independent instructions are removed before exact and semantic deduplication. Equivalent candidates are merged, whereas mechanistically distinct approaches are retained. Entries violating task constraints or failing an available native verifier are excluded.

Conventional-Solution Prompt

Identify the conventional approaches to the given task. Each approach must represent a distinct solution family rather than a paraphrase or minor implementation variant. For each family, provide its central principle, main reasoning or implementation steps, and applicable conditions. Do not propose deliberately unconventional solutions.
Task: {`task`}
Constraints: {`constraints`}

Figure 12: Prompt for eliciting additional conventional-solution candidates.

Quality-Anchor Prompt

Produce a complete response to the task below. Satisfy every explicit constraint and provide sufficient reasoning, implementation detail, or supporting evidence for independent evaluation. Do not refer to hidden references, evaluation criteria, or the data-construction process.
Task: {`task`}
Required output: {`response specification`}

Figure 13: Prompt for generating quality-anchor candidates from the public task.

The mechanism-atom set $\mathcal{P}(\mathcal{R}_{x_i})$ is extracted deterministically from each family's core approach, method, mechanism, resolution, and representative realization. Each representative solution contributes at most three mechanism atoms.

Quality-anchor bank. The bank $\mathcal{Q}_{x_i}$ contains complete and valid responses representing different acceptable realizations of a task. It provides positive quality references without imposing a unique solution. Suitable source references are used when available; otherwise, candidates are generated from the public task using the prompt in Figure 13.

An anchor must satisfy every essential quality criterion. Verifiable anchors must additionally pass the corresponding answer checker, execution environment, or constraint validator. Defective and near-duplicate anchors are excluded.

Validation and freezing. Both banks undergo structural validation, exact and semantic deduplication, and task-level review. Conventional families are checked for constraint compatibility, coverage, and mutual distinctness; anchors are checked for validity, completeness, and constraint satisfaction. Evaluation responses and test-time outputs are excluded from bank construction.

Each released entry records its task identifier, source, construction prompt, generator, verification result, semantic cluster, and version. The banks, encoder, normalization rules, mechanism extraction, calibration parameters, and Top-K configuration are frozen before formal evaluation.

A.10.6 Deterministic Metric

The deterministic Metric uses frozen verifiers and reference banks without an online language-model judge. It follows one scoring procedure rather than separate domain-level and instance-level runs.

Quality. For an objective verifier,

$$E_{i,Q}^{\text{metric}} = 10 v_i. \tag{33}$$

Otherwise, Quality is estimated from $\mathcal{Q}_{x_i}$:

$$E_{i,Q}^{\text{metric}} = 10 \,\text{clip}\left(\alpha_{d_i} \max_{a \in \mathcal{Q}_{x_i}} [0.4 f_{\text{sem}}(y_i, a) + 0.6 f_{\text{cov}}(y_i, a)] + \beta_{d_i}, 0, 1 \right). \tag{34}$$

Table 19: Domain-specific calibration for anchor-based Quality.

Calibration Group	α_d	β_d
Creative Writing	0.422499	0.353121
Mathematical Reasoning	0.622671	-0.243314
Physical Problem Solving	0.156200	0.254295
Scientific Ideation	0.120714	0.512086
Real-World Applications	0.274522	0.583764

Using the frozen Qwen3-Embedding-8B encoder Φ, the two features are

$$f_{\text{sem}}(y_i, a) = \text{clip}\left(\frac{\cos(\Phi(y_i), \Phi(a)) - \ell_{\text{sem}}}{u_{\text{sem}} - \ell_{\text{sem}}}, 0, 1\right),$$
$$f_{\text{cov}}(y_i, a) = \frac{|\mathcal{W}(y_i) \cap \mathcal{W}(a)|}{|\mathcal{W}(a)|}. \tag{35}$$

We set $\ell_{\text{sem}} = 0.35$ and $u_{\text{sem}} = 0.90$. The deterministic token set $\mathcal{W}(t)$ uses Unicode NFKC normalization and case folding, followed by alphanumeric tokenization for English and character bigrams for Chinese. No NLI or generative model is invoked.

Both features are computed against the same anchor before maximization, preventing evidence from unrelated anchors from being combined. Responses shorter than eight characters after normalization or matching an explicit refusal pattern receive zero Quality.

Only domains requiring anchor-based estimation are included in Table 19. For writing-choice tasks, an incorrect choice receives zero Quality; otherwise, the estimator is applied to the comparative rationale. Response length, structural size, direct response–rubric similarity, and cross-anchor feature composition are not used.

Novelty. A response is split deterministically at paragraph, sentence, semicolon, and list boundaries. Segments shorter than 12 characters are discarded, segments longer than 900 characters are divided, and at most eight segments are retained, yielding $\mathcal{S}_i$.

For $s \in \mathcal{S}_i$ and $p \in \mathcal{P}(\mathcal{R}_{x_i})$, define

$$\mathbf{h}_i = \text{norm}(\Phi(x_i)), \qquad \mathbf{u}_s = \Phi(s; \pi_{\text{mech}}), \qquad \mathbf{v}_p = \Phi(p). \tag{36}$$

To remove semantic content shared with the task, we project both mechanism representations away from the task direction:

$$\widetilde{\mathbf{u}}_s = \text{norm}(\mathbf{u}_s - \langle \mathbf{u}_s, \mathbf{h}_i \rangle \mathbf{h}_i),$$
$$\widetilde{\mathbf{v}}_p = \text{norm}(\mathbf{v}_p - \langle \mathbf{v}_p, \mathbf{h}_i \rangle \mathbf{h}_i). \tag{37}$$

The nearest conventional-mechanism similarity, calibrated distance, and task relevance are

$$m_s = \max_{p \in \mathcal{P}(\mathcal{R}_{x_i})} \langle \widetilde{\mathbf{u}}_s, \widetilde{\mathbf{v}}_p \rangle,$$
$$d_s = 1 - \text{clip}\left(\frac{m_s - \ell_{\text{conv}}}{u_{\text{conv}} - \ell_{\text{conv}}}, 0, 1\right), \tag{38}$$
$$\rho_s = \text{clip}\left(\frac{\cos(\Phi(s; \pi_{\text{rel}}), \Phi(x_i)) - \ell_{\text{rel}}}{u_{\text{rel}} - \ell_{\text{rel}}}, 0, 1\right).$$

We set $(\ell_{\text{conv}}, u_{\text{conv}}) = (-0.2, 0.6)$ and $(\ell_{\text{rel}}, u_{\text{rel}}) = (0.35, 0.90)$.

The fixed encoder instructions are shown in Figure 14.

Let $z_s = d_s \rho_s^2$ and $K_i = \min(2, |\mathcal{S}_i|)$. The final score is

$$E_{i,N}^{\text{metric}} = \frac{10}{K_i} \sum_{s \in \text{TopK}_{K_i}(\{z_s : s \in \mathcal{S}_i\})} z_s. \tag{39}$$

The relevance term suppresses unrelated additions, while Top-K averaging limits the influence of a single anomalous segment.

Encoder Instructions

Mechanism: Retrieve conventional solution mechanisms substantively used by the candidate. Focus on algorithms, proof ideas, causal mechanisms, designs, narrative realizations, object uses, and implementation principles; ignore wording and style.
Relevance: Given a task, retrieve candidate mechanism units that directly contribute to solving that task; ignore unrelated asides.

Figure 14: Fixed encoder instructions for mechanism and relevance retrieval.

A.10.7 Model-Based Evaluation Procedures

For dimension $k \in \{Q, N\}$ and mode $m \in \{\text{domain}, \text{instance}\}$, the evaluator receives a dimension-isolated context

$$\xi_{i,k}^{(m)} = \left(\tau_i, y_i, \mathcal{A}_{i,k}, \Pi_{i,k}^{(m)}\right), \tag{40}$$

where $\Pi_{i,k}^{(m)}$ contains only the criteria for dimension k. For Quality, $\mathcal{A}_{i,Q}$ contains applicable verifier evidence; for Novelty, $\mathcal{A}_{i,N}$ contains the conventional-solution bank. Scores and rationales from the other dimension are never included.

Strong Judge. A frontier evaluator directly produces every activated criterion score and its supporting evidence:

$$\mathbf{e}_{i,k}^{(m)} = g_{\text{s}}\left(\xi_{i,k}^{(m)}\right). \tag{41}$$

Decoding is deterministic where supported. Parent and overall scores are computed locally.

Trained Judge. The trained evaluator receives the same input and returns the same schema:

$$\mathbf{e}_{i,k}^{(m)} = g_{\text{t}}\left(\xi_{i,k}^{(m)}\right). \tag{42}$$

It is trained from Strong Judge supervision containing criterion-level scores and evidence. Generative checkpoints return structured scores directly; regression or classification checkpoints use their frozen training interface before conversion to the common 0–10 scale.

Review Committee. The committee contains three heterogeneous reviewers, an area chair, and the Strong Judge as program chair. For dimension k, reviewers first assess the response independently:

$$a_{r,k}^{(0)} = g_r\left(\xi_{i,k}^{(m)}\right), \qquad r = 1, 2, 3. \tag{43}$$

At round t, the area chair identifies agreements and unresolved claims, and reviewers reconsider only the disputed evidence:

$$\begin{aligned} b_k^{(t)} &= g_{\text{ac}}\left(\xi_{i,k}^{(m)}, \{a_{r,k}^{(t-1)}\}_{r=1}^{3}\right), \\ a_{r,k}^{(t)} &= g_r\left(\xi_{i,k}^{(m)}, b_k^{(t)}\right). \end{aligned} \tag{44}$$

Discussion terminates upon agreement or at the maximum number of rounds. The program chair then produces the final criterion scores from the complete record. Decisions are evidence-based rather than determined by majority vote.

Tool-Grounded Judge. A tool-using agent identifies claims requiring retrieval or execution and produces the trace

$$\mathcal{E}_{i,k}^{\text{tool}} = \left\{(q^{(t)}, o^{(t)})\right\}_{t=1}^{T_i^{\text{tool}}}, \tag{45}$$

where $q^{(t)}$ is an evidence request and $o^{(t)}$ the corresponding observation. Strong Judge evaluates

$$\mathbf{e}_{i,k}^{(m)} = g_{\text{s}}\left(\xi_{i,k}^{(m)}, \mathcal{E}_{i,k}^{\text{tool}}\right). \tag{46}$$

Retrieved precedents inform novelty, while execution and external evidence support correctness and feasibility. Failed retrieval is treated as inconclusive rather than evidence of originality.

Table 20: Maximum response lengths used in formal evaluation.

Domain	Maximum Tokens
Mathematical Reasoning	16,384
Code Generation	24,576
Commonsense Reasoning	4,096
Physical Problem Solving	4,096
Scientific Ideation	24,576
Creative Writing	16,384
Real-World Applications	16,384

Domain-Level Scoring Contract

Evaluate only the specified dimension using every supplied subdimension. Assign each subdimension a score from 0 to 10 and support it with response-specific evidence. Use only the evidence relevant to this dimension. Do not score any other dimension, Creative Space, or overall creativity.

Figure 15: Scoring-mode contract for the domain-level mode.

Adversarial Tournament. A challenger searches for factual errors, unmet constraints, unsupported claims, and conventional precedents; a defender responds using evidence consistent with the original answer:

$$\begin{aligned} u_k^{(t)} &= g_{\mathrm{c}}\left(\xi_{i,k}^{(m)}, \mathcal{E}_{i,k}^{(t-1)}\right), \\ p_k^{(t)} &= g_{\mathrm{d}}\left(\xi_{i,k}^{(m)}, u_k^{(t)}, \mathcal{E}_{i,k}^{(t-1)}\right), \\ \mathcal{E}_{i,k}^{(t)} &= \mathcal{E}_{i,k}^{(t-1)} \cup \{(u_k^{(t)}, p_k^{(t)})\}. \end{aligned} \tag{47}$$

Strong Judge adjudicates the complete record and scores the original response, not the rhetorical performance of either agent.

The candidate is additionally compared with representative conventional responses and a stronger adversarial response. Order-balanced arbiters inspect both presentation orders, while prediction, precedent, and grounding probes attempt to falsify apparent novelty. Pairwise evidence is aggregated before the shared scoring rule is applied. Creative Space does not determine pairwise outcomes.

A.10.8 Response Generation and Replay

The evaluated model is called once per task with reasoning output disabled. The stored response is replayed unchanged across all applicable methods and scoring modes. The frozen generation profile uses $\mathrm{temperature} = 0.7$, $\mathrm{top_p} = 0.8$, $\mathrm{top_k} = 20$, presence penalty 1.5, and seed 42.

Unknown termination states and unintended truncation fail the release check. Raw responses are retained for source-benchmark parsers, while CreativitySuite evaluators consume a deterministic canonical representation. The adapter may remove protocol-specific wrappers but cannot repair substantive content or invalid native outputs.

A.10.9 Evaluation Prompt Contracts

All formal prompts and evaluation artifacts are in English. Quality and Novelty are evaluated through independent calls. Each call is explicitly instructed not to infer or score the other dimension.

Scoring-mode contract. The domain-level mode uses the corresponding subdimensions in Table 6 with the contract in Figure 15.

The instance-level mode uses task-specific rubrics with the contract in Figure 16.

Strong and Trained Judges. Strong Judge receives the appropriate scoring-mode contract together with the prompt in Figure 17.

Instance-Level Scoring Contract

Evaluate only the specified dimension. Score every active rubric item against the candidate response, without exceeding its stated maximum, and cite the supporting evidence. Do not omit an active item or score any other dimension, Creative Space, or overall creativity.

Figure 16: Scoring-mode contract for the instance-level mode.

Strong Judge Prompt

Act as an impartial evaluator. Assess only the requested dimension using the task, response, dimension-specific evidence, and supplied criteria. Reward substantive validity or originality rather than fluency, length, or lexical variation. Return all required criterion scores with concise evidence.

Figure 17: Strong Judge prompt, used together with the scoring-mode contract.

Trained Judge follows the same input and output contract. Its supervision contains the criterion-level scores and supporting evidence produced by Strong Judge.

Review Committee. Each reviewer receives the prompt in Figure 18.

The area chair receives the prompt in Figure 19.

The program chair receives the prompt in Figure 20.

Tool-Grounded Judge. The tool agent receives the prompt in Figure 21.

Strong Judge then receives the prompt in Figure 22.

Adversarial Tournament. The challenger receives the prompt in Figure 23.

The defender receives the prompt in Figure 24.

The adjudicator receives the prompt in Figure 25.

The deterministic Metric has no online evaluation prompt. Its offline conventional-solution and quality-anchor prompts are specified in Appendix A.10.5.

A.10.10 Output Validation and Reproducibility

A valid model-based evaluation must contain every required domain subdimension or activated rubric item. Missing fields, non-numeric values, out-of-range scores, and incomplete rubric coverage are not imputed. Agentic methods may issue one targeted format-repair request without adding new evidence or revising the substantive judgment; unresolved outputs are recorded as evaluation failures.

Each result records the protocol identifier, requested dimension, prompt language, scoring mode, evaluator configuration, response identifier, verifier route and result, reference-bank versions, prompt template and hash, decoding configuration, raw evaluator output, parsed scores, coverage status, and final aggregation result. Domain-level and instance-level results remain separate.

Formal evaluation freezes the task and response identifiers, evaluator versions, prompts, verifier assets, reference banks, embedding encoder, normalization rules, calibration parameters, and aggregation weights. Representative traces and complete method-specific prompts are released with the evaluation artifacts.

A.11 Effectiveness of Real-World Creativity Training Data

Setup. We perform full-parameter SFT on Qwen2.5-32B-Instruct using 7,969 Real-World creativity examples. For each task, the training target is selected from four teacher responses through

Reviewer Prompt

Evaluate the requested dimension independently. Report all criterion-level scores, supporting evidence, and uncertainty. Do not infer the opinions of other reviewers or assess any unrequested dimension.

Figure 18: Review Committee: reviewer prompt.

Area Chair Prompt

Compare the reviews and separate substantive agreements from unresolved disagreements. If all required judgments agree, terminate deliberation. Otherwise, issue targeted evidence-based questions without assigning final scores.

Figure 19: Review Committee: area-chair prompt.

blinded ranking. The model is trained for three epochs in BF16 with an effective batch size of 64 and a learning rate of 1×10^{-5}.

We compare the base and SFT models on a fixed set of 500 creativity problems using identical prompts and decoding parameters. Responses are generated once and evaluated on a 0–10 scale using instance-specific rubrics.

Results. SFT substantially improves all evaluated dimensions and outperforms the base model on every paired example, demonstrating the effectiveness of the Real-World creativity training data under the fixed evaluation protocol.

Discussion. SFT also produces more complete responses. Because the detailed rubrics reward requirement coverage, the observed improvement reflects gains in both content quality and response completeness.

A.12 Training Hyperparameters and Implementation Details

Common settings. All reported models are trained with full-parameter supervised fine-tuning using LLaMA-Factory, PyTorch, and DeepSpeed. Training uses a maximum sequence length of 8,192 tokens without sequence packing, and loss is computed only over assistant responses. Unless otherwise specified, we train for one epoch with a global batch size of 16, AdamW, a cosine learning-rate schedule, a 3% warm-up ratio, weight decay of 0.01, and gradient clipping at 1.0. We use BF16, TF32, gradient checkpointing, and a fixed random seed of 42. Training is conducted on up to eight NVIDIA L20X GPUs.

Single-domain SFT. The single-domain experiments are conducted on the Qwen models using the complete training set of each domain. Table 22 reports the dataset sizes, peak learning rates, and validation-selected checkpoints. All runs use a global batch size of 16 and DeepSpeed ZeRO-2. Checkpoints are selected exclusively on the corresponding frozen development set; test examples are never used for model selection.

Seven-domain SFT. The seven-domain training set contains 19,858 examples: 4,015 mathematics, 2,161 physical, 2,052 code, 2,209 science, 3,303 writing, 2,261 commonsense, and 3,857 Real-World Applications examples. The domain datasets are concatenated without resampling or reweighting. Table 23 summarizes the model-specific settings. All runs use one epoch, a global batch size of 16, and a per-device batch size of one.

Model-specific implementation. Qwen models use FlashAttention-2 and their native chat templates. Gemma models use SDPA in text-only mode; their vision tower and multimodal projector remain frozen while all language-model parameters are updated. We remove explicit `<think>` blocks from the Gemma training view and disable thinking during training. Four checkpoints are

Program Chair Prompt

Adjudicate the requested dimension from the complete review record. Resolve disagreements using task evidence and verifier results rather than majority vote alone. Return the final criterion-level scores with concise justifications.

Figure 20: Review Committee: program-chair prompt.

Tool Agent Prompt

Identify claims relevant to the requested dimension that require external evidence. Use the available retrieval or execution tools and record each request and observation. Distinguish supporting, contradicting, and inconclusive evidence. Do not treat failed retrieval as proof of originality.

Figure 21: Tool-Grounded Judge: tool-agent prompt.

retained at approximately 25%, 50%, 75%, and 100% of training. Final checkpoints are selected using only frozen development data, and each corresponding test set is evaluated once.

A.13 Additional Domain-Specific Results for Gemma

We additionally evaluate Gemma-4-12B and Gemma-4-31B before and after all-domain SFT. Both models are trained on the same 19,858 seven-domain samples. Checkpoints are selected using development results, without access to the test sets, and evaluated under the original protocols of the six external benchmarks.

All-domain SFT produces heterogeneous effects across domains and model scales. For Gemma-4-12B, improvements in physical problem solving and scientific ideation are offset by lower mathematical novelty, code performance, and writing accuracy, resulting in a slightly lower macro-average. Gemma-4-31B shows a modest aggregate improvement, with gains in mathematics, commonsense reasoning, and scientific ideation, but small regressions in code generation and mixed changes on physical problem solving.

These results reinforce that joint SFT does not uniformly improve creativity. Instead, it redistributes performance across domains and evaluation dimensions, with larger models appearing less susceptible to negative transfer. Because no significance tests are conducted for these external benchmarks, the comparisons are descriptive.

A.14 In-depth Analysis

A.14.1 Effect of Creative Space

We examine whether Creative Space Ω captures opportunities for producing valid novel solutions and calibrates task value in cross-instance evaluation. All Ω scores are determined before response generation and do not use candidate responses.

Operationalization. Under the *one-path prior*, each valid combination of a conventional solution family and a task-relevant gain type constitutes one creative path:

$$n_i = \sum_{r \in \mathcal{R}_{x_i}} \sum_{t \in \widetilde{\mathcal{T}}_{x_i}} \mathbb{I}_i(r,t), \qquad \Omega_i = 10 \frac{\log(n_i + 2)}{\log 38}, \tag{48}$$

where $\mathbb{I}_i(r,t) = 1$ only if the combination yields a concrete, valid, and substantively distinct solution. The logarithm reflects diminishing information from additional paths. For the same task, Ω_i is fixed across responses and models.

Cross-Instance Analysis. We stratify tasks by Ω within each of seven domains and sample 20 Low-, Mid-, and High-Ω tasks per domain, yielding 420 tasks and 1,260 model–task observations

Tool-Grounded Strong Judge Prompt

Evaluate only the requested dimension using the task, response, active criteria, and complete tool trace. Prefer reproducible evidence to unsupported assertions, and treat missing evidence as uncertainty rather than automatic failure.

Figure 22: Tool-Grounded Judge: Strong Judge prompt with the complete tool trace.

Challenger Prompt

Challenge claims relevant to the requested dimension. Identify specific errors, unmet constraints, unsupported claims, conventional precedents, or stronger alternatives, grounding each challenge in the task or available evidence.

Figure 23: Adversarial Tournament: challenger prompt.

for Qwen3.5-4B, Qwen3.5-9B, and Qwen3.8-27B. Tied Ω values are assigned to the same stratum; the mean scores of the three strata are 4.979, 6.417, and 7.724.

Quality and Novelty are evaluated separately using order-balanced comparisons. Confidence intervals are obtained from 10,000 domain-stratified, task-clustered bootstrap samples. As shown in Figure 28(a), all three models achieve higher Novelty on High-Ω tasks than on Low-Ω tasks. The High–Low differences are $+0.713$, $+0.602$, and $+0.584$, respectively, although only the Qwen3.5-4B interval excludes zero. Quality exhibits no consistent relationship with Ω. After restricting the analysis to responses meeting the primary quality threshold $Q \geq 6$, the Novelty effects remain positive but uncertain. Thus, the cross-instance analysis provides suggestive, rather than causal, evidence that broader creative spaces support more novel responses.

Within-Instance Controlled Experiment. We construct paired Low/High-Ω versions of 100 tasks from physical problem solving, scientific ideation, creative writing, and real-world applications. Both versions preserve the task objective, background, knowledge requirements, quality criteria, output format, and requirement to submit one primary solution. The Low-Ω version requires a specified conventional mechanism, whereas the High-Ω version permits other valid mechanisms.

An independent manipulation check verifies that only the admissible solution space changes. Ninety-three pairs pass: 23 physical, 25 scientific, 25 writing, and 20 real-world tasks. Each version is sampled four times using Qwen3.5-4B, Qwen3.5-9B, and Qwen3.8-27B, yielding 2,232 responses. Quality and Novelty are evaluated separately in both presentation orders. All 4,464 evaluation calls succeed, with mirrored-order agreement of 68.82% for Quality and 84.77% for Novelty. Confidence intervals use 10,000 task-clustered bootstrap samples that keep both versions, all models, and all generations of the same task together.

Expanding the admissible solution space significantly increases Novelty for all three models, by 2.51, 2.55, and 3.12 points, respectively, while mean Quality decreases by 0.60, 1.12, and 0.57 points (Figure 28(b)). This indicates that additional creative opportunities also introduce mechanism-selection and execution risk.

We define joint success as $Q \geq 6$ and $N \geq 6$. Its rate increases by 34.95, 32.26, and 46.51 percentage points for Qwen3.5-4B, Qwen3.5-9B, and Qwen3.8-27B, respectively (Figure 28(c)). When restricted to paired responses meeting $Q \geq 6$ under both conditions, Novelty still increases by 3.01, 2.58, and 3.07 points. The observed gain therefore reflects valid mechanism-level innovation rather than low-quality divergence.

The Qwen3.8-27B model benefits more than the two smaller models in Novelty and joint success, whereas Qwen3.5-4B and Qwen3.5-9B show no reliable difference in these effects. Across domains, gains are largest for scientific ideation and real-world applications and more modest for creative writing. Physical problem solving exhibits the clearest Quality decline, suggesting greater difficulty in maintaining feasibility when exploring alternative mechanisms. These domain-level comparisons are descriptive because separate clustered confidence intervals are not estimated for each domain.

Defender Prompt

Address each challenge using evidence consistent with the original response. Correct misunderstandings and defend supported claims, but concede limitations that cannot be justified. Do not introduce claims absent from or inconsistent with the original response.

Figure 24: Adversarial Tournament: defender prompt.

Adjudicator Prompt

Evaluate only the requested dimension of the original response in light of the complete challenge–defense record. Score the response rather than the rhetorical performance of either agent, and determine whether its claims survive the identified counterexamples or conventional alternatives.

Figure 25: Adversarial Tournament: adjudicator prompt.

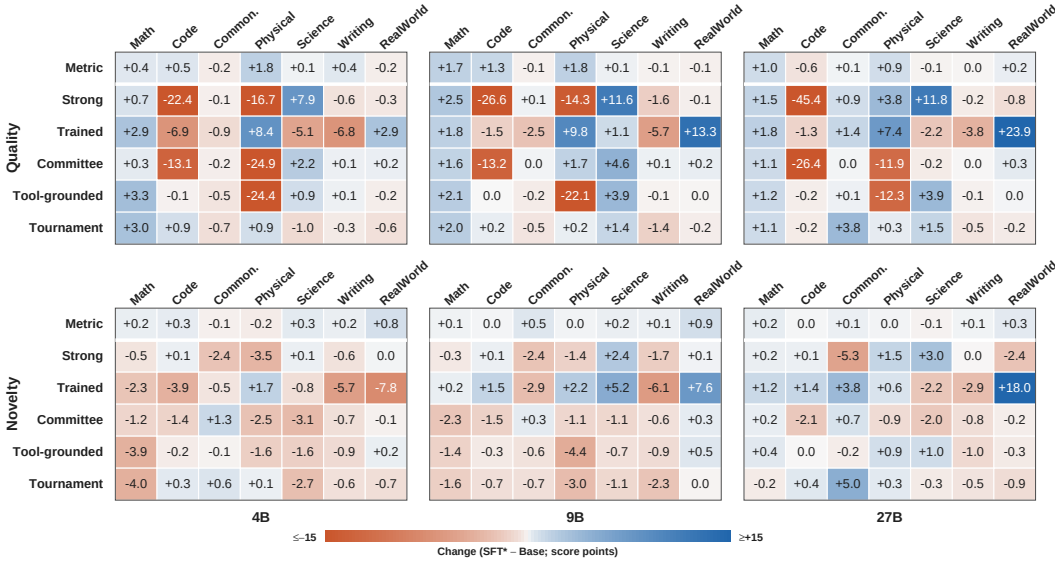

Figure 26: Domain-level changes in quality and novelty after full-domain SFT across seven domains and three model scales.

Cross-Instance Value Calibration. To isolate the effect of Ω, we first aggregate Quality and Novelty while preserving their relative weights:

$$C_i^{QN} = 10 \left(\frac{Q_i}{10}\right)^{\frac{w_Q}{w_Q+w_N}} \left(\frac{N_i}{10}\right)^{\frac{w_N}{w_Q+w_N}}. \tag{49}$$

Including Creative Space gives

$$C_i^{QN\Omega} = 10 \left(\frac{C_i^{QN}}{10}\right)^{1-w_\Omega} \left(\frac{\Omega_i}{10}\right)^{w_\Omega}. \tag{50}$$

Because Ω_i is shared by all responses to task i, this transformation preserves within-task model rankings while adjusting the task's contribution to cross-instance aggregation.

As shown in Figure 28(d), High-Ω tasks receive 2.22–2.81 points more adjustment than Low-Ω tasks across the three models. Consequently, the aggregate contribution of the Low group decreases by 0.83–0.95 percentage points, while that of the High group increases by 0.78–0.93 percentage points. This calibration is not designed to maximize leaderboard separation: depending on where model differences occur, it may widen, preserve, or narrow aggregate performance gaps.

Table 21: Effect of Real-World creativity SFT. W/L/T denotes the paired win/loss/tie counts of SFT against the base model.

Model	Overall	Novelty	Feasibility	Audience	Median Length	W/L/T
Base	2.165	1.948	3.921	3.012	1,063	–
SFT	**8.611**	**9.338**	**8.678**	**9.203**	2,547	500/0/0
Δ	+6.446	+7.390	+4.757	+6.191	+1,484	–

Table 22: Single-domain SFT configurations. Each learning-rate and checkpoint tuple follows the order 4B/9B/27B.

Domain	Training examples	Peak learning rate (4B/9B/27B)	Selected step (4B/9B/27B)
Mathematics	4,015	$5\times10^{-7}/3\times10^{-7}/1\times10^{-7}$	62/124/251
Physical	2,161	$1\times10^{-6}/5\times10^{-7}/2\times10^{-7}$	34/102/136
Code	2,052	$2.5\times10^{-8}/2.5\times10^{-8}/2.5\times10^{-8}$	129/64/64
Science	2,209	$5\times10^{-7}/3\times10^{-7}/3\times10^{-8}$	70/105/70
Writing	3,303	$5\times10^{-7}/3\times10^{-7}/1\times10^{-7}$	204/153/102
Commonsense	2,261	$5\times10^{-7}/3\times10^{-7}/1\times10^{-7}$	35/142/70

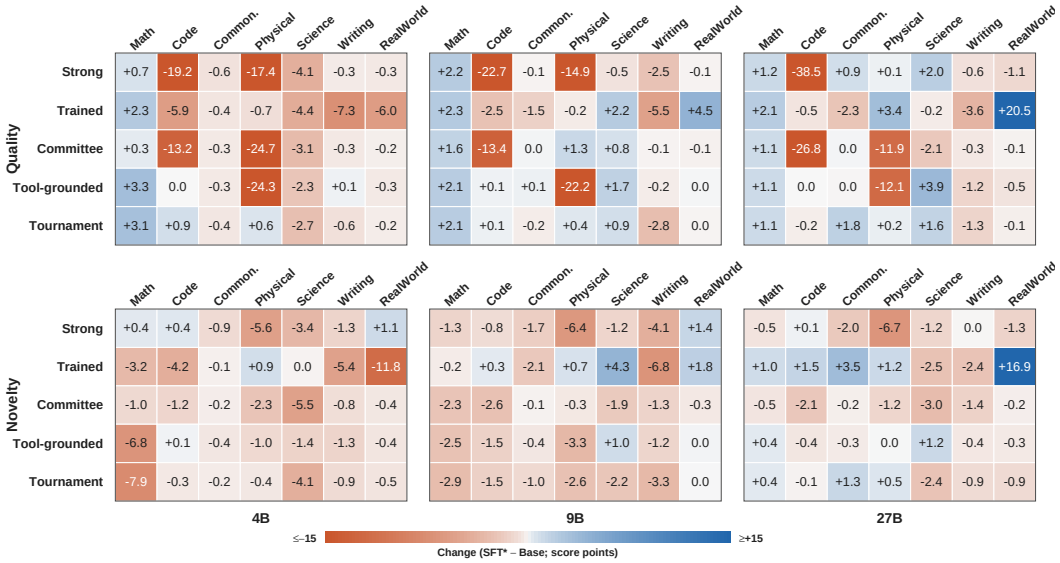

Figure 27: Instance-level changes in quality and novelty after full-domain SFT across seven domains and three model scales.

Evidence Scope. The controlled experiment provides stronger evidence than the cross-instance analysis but covers only four domains. It tests whether models exploit an explicitly expanded solution space, and the observed Quality decline may reflect both exploration risk and additional mechanism-selection burden. Moreover, manipulation checks and response evaluation rely on model judges, with lower mirrored agreement for Quality than Novelty. The reweighting analysis verifies the aggregation behavior of Ω but does not by itself establish the accuracy of every task-level annotation.

Together, the results support two roles for Creative Space: it characterizes the opportunity to produce valid novel solutions and calibrates the relative value of tasks in cross-instance evaluation. It complements, rather than replaces, response-side Quality and Novelty.

Table 23: Model-specific configurations for seven-domain SFT.

Model	Peak learning rate	GPUs	DeepSpeed
Qwen3.5-4B	2.5×10^{-8}	8	ZeRO-2
Qwen3.5-9B	3×10^{-8}	8	ZeRO-2
Qwen3.8-27B	3×10^{-8}	8	ZeRO-2
Gemma-4-E4B	3×10^{-7}	4	ZeRO-2
Gemma-4-12B	3×10^{-7}	4	ZeRO-3
Gemma-4-31B	3×10^{-8}	8	ZeRO-3

Table 24: Domain-specific creativity results for Gemma models. SFT* denotes SFT on all seven domains. ***Avg.*** is the macro-average over the six benchmarks after mapping metrics to $[0, 100]$ and averaging the displayed metrics within each benchmark. The better result within each model scale is shown in bold.

Model	**Method**	**CreativeMath**		**CreativeBench**		**CresOWLve**	**CreativityBench**		**LiveIdeaBench**	**CreataSet**	***Avg.***
		Correctness ↑	Novelty ↑	Pass@1 ↑	Creativity ↑	Accuracy ↑	Entity Correct ↑	Gold Correct ↑	Average ↑	Accuracy ↑	
Gemma-4-12B	Base	**58.27**	**18.20**	**45.69**	**0.3060**	**5.01**	43.74	7.19	6.0785	**74.77**	**40.40**
	SFT*	58.09	14.71	37.19	0.2443	4.69	**52.96**	**9.39**	**6.5682**	70.44	39.87
Gemma-4-31B	Base	63.24	24.08	**53.29**	**0.3574**	10.78	**40.53**	15.64	6.3529	74.20	44.13
	SFT*	**64.52**	**24.45**	52.87	0.3544	**11.16**	40.26	**15.91**	**6.4288**	**74.24**	**44.40**

A.14.2 Validity of the Deterministic Metric

We evaluate whether the frozen deterministic metric recovers reference quality preferences and detects mechanism-level novelty. All formulas and calibration parameters are fixed before these confirmatory experiments.

Quality Ranking. The quality experiment samples 500 questions from each of five domains, yielding 2,500 questions without native verifiers. We evaluate frozen responses from 4B and 9B Base and SFT models. For response i, let t_i^{domain} and t_i^{instance} denote Quality scores from independently executed domain-level and instance-level Strong Judge evaluations. A response is retained when

$$\left|t_i^{\text{domain}} - t_i^{\text{instance}}\right| \le \tau, \qquad t_i = \frac{t_i^{\text{domain}} + t_i^{\text{instance}}}{2}. \tag{51}$$

The main setting uses $\tau = 0.10$, balancing agreement between the two scoring modes with sample coverage. It retains 6,165 responses to 2,206 questions and forms 1,772 same-question pairs with distinct reference scores.

For the set $\mathcal{P}_\tau$ of pairs satisfying $t_{i,a} > t_{i,b}$, pairwise agreement is

$$\text{Acc}_Q(\tau) = \frac{1}{|\mathcal{P}_\tau|} \sum_{(i,a,b)\in\mathcal{P}_\tau} \mathbb{I}\left[E_{i,Q}^{\text{metric}}(y_{i,a}) > E_{i,Q}^{\text{metric}}(y_{i,b})\right]. \tag{52}$$

Comparing responses to the same question controls for question difficulty. As shown in Figure 29(a), the metric agrees with the reference ordering on 62.81% of pairs. Its question-level bootstrap 95% confidence interval, $[60.13, 65.43]$, lies above the 50% random-ranking baseline. Every domain has a point estimate above 50%, although performance varies across domains.

The result is insensitive to the consensus threshold (Figure 29(b)). Thresholds of 0.05, 0.10, and 0.20 retain respectively 657, 1,772, and 3,494 pairs, with agreement rates of 62.79%, 62.81%, and 63.09%. Using all dual-scored responses yields 61.69%, while domain-level and instance-level labels alone yield 59.21% and 60.26%. Thus, consensus labels provide a more stable reference than either scoring mode alone.

Novelty Discrimination. The novelty experiment uses 3,500 disjoint questions, with 500 per domain. Each question has five controlled responses: a conventional solution, a surface paraphrase preserving its mechanism, a novel mechanism, a mixed response combining conventional and new mechanisms, and an irrelevant hard negative. The resulting 17,500 responses isolate mechanism novelty, surface variation, and unrelated additions.

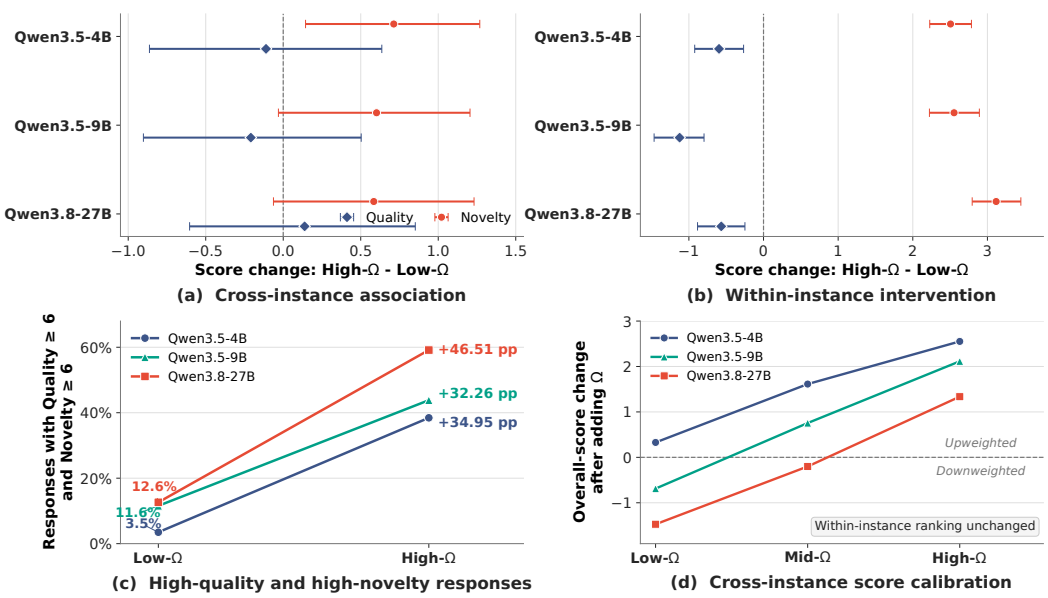

Figure 28: Effects of Creative Space on Quality, Novelty, high-quality novel responses, and cross-instance score calibration.

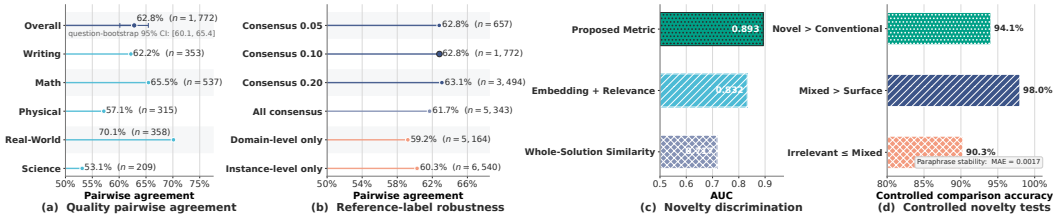

Figure 29: Empirical validity of the deterministic metric. **(a)** Pairwise agreement between quality scores and reference assessments across domains; the error bar denotes the 95% confidence interval for the overall result. **(b)** Robustness of quality ranking to alternative reference-label constructions. **(c)** Novelty discrimination against semantic similarity baselines. **(d)** Controlled evaluation of mechanism sensitivity, irrelevant-content suppression, and paraphrase stability.

As shown in Figure 29(c), the proposed metric achieves an AUC of 0.893 in distinguishing Novel from Conventional responses, compared with 0.832 for *Embedding + Relevance* and 0.717 for *Whole-Solution Similarity*. The former retains mechanism segmentation, task relevance, and Top-2 aggregation but omits projection away from the task direction. The latter directly compares complete responses with conventional solutions and therefore does not isolate local mechanisms or irrelevant content. Their lower AUCs support the use of task-conditioned, mechanism-level comparison.

Controlled comparisons further test the intended behavior (Figure 29(d)). The metric ranks Novel above Conventional in 94.06% of questions and Mixed above Surface in 97.99%, demonstrating sensitivity to substantive mechanisms rather than wording changes. It also ranks Irrelevant no higher than Mixed in 90.26%, indicating that task-relevance weighting suppresses spurious novelty.

For a response y_i and a mechanism-preserving paraphrase $\widetilde{y}_i$, paraphrase stability is measured on the normalized $[0, 1]$ scale:

$$\mathrm{MAE}_{\mathrm{para}} = \frac{1}{n} \sum_{i=1}^{n} \left| \overline{N}_i^{\mathrm{metric}}(y_i) - \overline{N}_i^{\mathrm{metric}}(\widetilde{y}_i) \right|. \tag{53}$$

The resulting MAE is 0.0017, corresponding to only 0.017 points on the $[0, 10]$ scale. The novelty score is therefore largely invariant to mechanism-preserving paraphrases.

Overall, the deterministic metric recovers reference Quality rankings and distinguishes substantive mechanism changes without online model judgment. Quality validation establishes agreement with

stable Strong Judge references, rather than human ground truth, while Novelty remains relative to the coverage of the frozen conventional-solution bank $\mathcal{R}_x$.

A.15 Cross-Domain Evaluation Protocols and Detailed Results

Evaluation protocol. All evaluators score the same frozen model responses. The five model-based evaluators operate in two modes: domain-level evaluation uses criteria shared within each domain, whereas instance-level evaluation uses question-specific rubrics. The Deterministic Metric follows a single frozen protocol.

Quality measures correctness, validity, and task fulfillment, while Novelty measures substantive originality relative to conventional solutions. Creative Space is a frozen question-level property shared by Base and SFT*. For verifier-backed tasks, objective correctness contributes 60% of Quality and the remaining criteria contribute 40%.

For all methods except the Trained Judge, creativity is computed as

$$C_i = 10 \prod_{k \in \{Q,N,\Omega\}} \left(\frac{S_{i,k}}{10} \right)^{w_{d,k}}, \qquad \sum_k w_{d,k} = 1, \tag{54}$$

where Q, N, and Ω denote Quality, Novelty, and Creative Space. The weights are $(0.50, 0.40, 0.10)$ for Math, Code, Commonsense, and Physical; $(0.45, 0.40, 0.15)$ for Science and Writing; and $(0.40, 0.40, 0.20)$ for RealWorld.

The Deterministic Metric combines task-native verifiers with frozen quality and novelty proxies. The Strong Judge evaluates Quality and Novelty independently. The Review Committee aggregates multiple structured reviews, the Tool-Grounded Judge evaluates novelty against retrieved prior art, and the Adversarial Tournament uses a challenger with separate Quality and Novelty arbiters.

The Trained Judge is a frozen Bradley–Terry reward model. Its raw logits are converted into paired Base–SFT* percentile ranks, r_i^Q and r_i^N, and its creativity utility is

$$U_i = \frac{\Omega_i}{10} r_i^Q \left(0.7 + 0.3 r_i^N\right). \tag{55}$$

These values are relative ranks rather than calibrated absolute scores and should not be compared directly with other evaluators.

Detailed numerical results. Tables 25–28 report the complete Base and SFT* Quality and Novelty scores. Under the Deterministic Metric, the macro-average Quality scores change from 35.46 to 35.86, 36.00 to 36.67, and 39.33 to 39.55 for the 4B, 9B, and 27B models, respectively; the corresponding Novelty scores change from 3.69 to 3.89, 3.91 to 4.17, and 4.03 to 4.14.

Mathematical Quality increases consistently. For example, the domain-level Strong Judge scores increase from 13.93 to 14.59, 16.79 to 19.32, and 19.53 to 21.07. In contrast, its Code Quality scores decrease from 34.49 to 12.08, 39.24 to 12.62, and 59.07 to 13.68. Writing Novelty also generally decreases under model-based evaluation. Results for Science, Physical, and RealWorld vary more substantially across evaluators and model scales.

Because CreativitySuite contains refined instances from heterogeneous external benchmarks, absolute scores are not directly comparable across evaluators or domains. We therefore interpret Base–SFT* comparisons within each evaluator and report all findings descriptively. Additionally, Figures 26 and 27 show heatmaps of domain-level and instance-level patterns in quality and novelty changes, respectively.

Table 25: Detailed Quality scores under the deterministic and domain-level evaluation protocols. Each row reports the absolute Base or SFT* score. ***Avg.*** is the macro-average over the seven domains. Trained Judge values are paired percentile ranks rather than calibrated absolute scores. All values are rounded to two decimal places.

Evaluator	Model	Setting	Math	Code	Common.	Physical	Science	Writing	RealWorld	*Avg.*
Deterministic Metric	4B	Base	5.39	28.00	2.47	49.16	52.94	38.93	71.31	**35.46**
		SFT*	5.82	28.50	2.25	50.99	53.02	39.33	71.14	**35.86**
	9B	Base	4.62	32.86	3.51	46.02	52.87	40.64	71.48	**36.00**
		SFT*	6.37	34.20	3.40	47.79	52.99	40.55	71.36	**36.67**
	27B	Base	6.23	56.16	5.21	39.75	53.02	43.14	71.79	**39.33**
		SFT*	7.24	55.60	5.26	40.68	52.96	43.11	71.98	**39.55**
Domain-level evaluation										
Strong Judge	4B	Base	13.93	34.49	2.05	43.18	42.12	74.25	75.07	**40.73**
		SFT*	14.59	12.08	1.91	26.45	50.04	73.68	74.72	**36.21**
	9B	Base	16.79	39.24	2.86	42.56	42.39	77.59	78.65	**42.87**
		SFT*	19.32	12.62	2.95	28.31	54.03	75.95	78.51	**38.81**
	27B	Base	19.53	59.07	4.29	41.18	49.44	80.98	81.82	**48.04**
		SFT*	21.07	13.68	5.21	44.95	61.20	80.74	81.03	**43.98**
Trained Judge	4B	Base	48.57	53.43	50.45	45.82	52.57	53.42	48.56	**50.40**
		SFT*	51.43	46.57	49.55	54.18	47.43	46.58	51.44	**49.60**
	9B	Base	49.10	50.75	51.23	45.12	49.46	52.86	43.35	**48.84**
		SFT*	50.90	49.25	48.77	54.88	50.54	47.14	56.65	**51.16**
	27B	Base	49.08	50.66	49.31	46.30	51.08	51.90	38.05	**48.05**
		SFT*	50.92	49.34	50.69	53.70	48.92	48.10	61.95	**51.95**
Review Committee	4B	Base	4.97	23.35	1.86	42.79	54.49	65.88	73.36	**38.10**
		SFT*	5.27	10.29	1.70	17.86	56.64	66.02	73.52	**33.04**
	9B	Base	4.23	26.83	2.79	40.96	55.87	69.25	75.88	**39.40**
		SFT*	5.84	13.58	2.84	42.62	60.47	69.37	76.09	**38.69**
	27B	Base	5.62	46.43	4.09	36.09	61.79	72.38	77.68	**43.44**
		SFT*	6.72	20.04	4.13	24.19	61.55	72.37	77.98	**38.14**
Tool-Grounded Judge	4B	Base	5.93	0.09	13.98	41.99	53.95	65.59	69.54	**35.87**
		SFT*	9.27	0.00	13.49	17.57	54.89	65.70	69.36	**32.90**
	9B	Base	5.23	0.00	16.46	40.25	55.20	69.57	73.06	**37.11**
		SFT*	7.33	0.00	16.24	18.12	59.09	69.51	73.03	**34.76**
	27B	Base	7.02	0.25	16.00	35.62	62.31	72.99	74.64	**38.40**
		SFT*	8.19	0.08	16.11	23.29	66.16	72.89	74.67	**37.34**
Adversarial Tournament	4B	Base	5.98	20.31	18.68	24.70	32.72	48.04	63.21	**30.52**
		SFT*	8.98	21.16	17.95	25.59	31.76	47.77	62.56	**30.83**
	9B	Base	5.31	24.67	22.51	25.92	33.31	51.65	65.84	**32.74**
		SFT*	7.33	24.88	22.04	26.10	34.72	50.21	65.65	**32.99**
	27B	Base	7.22	43.12	20.99	25.18	38.84	54.52	68.67	**36.93**
		SFT*	8.27	42.89	24.78	25.43	40.32	54.03	68.42	**37.73**

Table 26: Detailed Novelty scores under the deterministic and domain-level evaluation protocols. Each row reports the absolute Base or SFT* score. ***Avg.*** is the macro-average over the seven domains. Trained Judge values are paired percentile ranks rather than calibrated absolute scores. All values are rounded to two decimal places.

Evaluator	Model	Setting	Math	Code	Common.	Physical	Science	Writing	RealWorld	*Avg.*
Deterministic Metric	4B	Base	3.14	5.40	1.29	4.60	2.30	2.68	6.42	**3.69**
		SFT*	3.32	5.65	1.18	4.38	2.63	2.84	7.24	**3.89**
	9B	Base	3.96	5.88	2.12	4.39	2.12	2.55	6.35	**3.91**
		SFT*	4.03	5.90	2.59	4.43	2.30	2.69	7.23	**4.17**
	27B	Base	6.88	5.89	0.47	3.48	1.92	2.73	6.87	**4.03**
		SFT*	7.13	5.90	0.61	3.52	1.85	2.79	7.19	**4.14**
Domain-level evaluation										
Strong Judge	4B	Base	7.92	10.38	20.32	34.96	27.65	46.85	40.10	**26.88**
		SFT*	7.42	10.53	17.92	31.44	27.80	46.27	40.13	**25.93**
	9B	Base	8.10	10.45	21.75	34.28	28.59	49.64	42.54	**27.91**
		SFT*	7.76	10.59	19.38	32.85	31.02	47.91	42.60	**27.45**
	27B	Base	6.96	10.73	28.46	32.56	30.74	48.88	43.83	**28.88**
		SFT*	7.13	10.88	23.17	34.04	33.74	48.87	41.42	**28.46**
Trained Judge	4B	Base	51.16	51.94	50.23	49.16	50.38	52.87	53.92	**51.38**
		SFT*	48.84	48.06	49.77	50.84	49.62	47.13	46.08	**48.62**
	9B	Base	49.91	49.23	51.43	48.91	47.38	53.06	46.20	**49.45**
		SFT*	50.09	50.77	48.57	51.09	52.62	46.94	53.80	**50.55**
	27B	Base	49.40	49.30	48.11	49.69	51.11	51.43	40.99	**48.57**
		SFT*	50.60	50.70	51.89	50.31	48.89	48.57	59.01	**51.43**
Review Committee	4B	Base	29.98	15.03	3.31	35.05	49.76	57.96	66.97	**36.87**
		SFT*	28.77	13.68	4.56	32.56	46.63	57.25	66.87	**35.76**
	9B	Base	28.29	16.33	5.24	36.44	51.35	60.27	69.10	**38.14**
		SFT*	26.01	14.86	5.55	35.30	50.23	59.63	69.37	**37.28**
	27B	Base	25.23	16.92	3.51	36.49	53.57	59.99	70.33	**38.01**
		SFT*	25.44	14.87	4.19	35.60	51.53	59.24	70.16	**37.29**
Tool-Grounded Judge	4B	Base	18.89	12.36	10.36	37.30	38.06	43.98	54.22	**30.74**
		SFT*	14.94	12.13	10.30	35.71	36.50	43.05	54.40	**29.58**
	9B	Base	18.59	12.73	12.84	38.21	38.85	47.00	56.60	**32.12**
		SFT*	17.19	12.44	12.27	33.83	38.19	46.12	57.06	**31.01**
	27B	Base	16.60	12.08	12.25	37.21	43.67	44.91	59.20	**32.27**
		SFT*	16.97	12.08	12.04	38.11	44.67	43.92	58.93	**32.39**
Adversarial Tournament	4B	Base	28.47	29.78	29.69	41.49	33.19	52.44	60.49	**39.36**
		SFT*	24.44	30.13	30.25	41.63	30.45	51.80	59.74	**38.35**
	9B	Base	29.20	30.88	31.48	43.77	34.16	54.37	62.73	**40.94**
		SFT*	27.60	30.14	30.80	40.81	33.08	52.05	62.70	**39.60**
	27B	Base	26.87	30.21	29.42	45.03	37.46	51.45	62.86	**40.47**
		SFT*	26.63	30.61	34.42	45.37	37.19	50.92	62.00	**41.02**

Table 27: Detailed Quality scores under instance-level evaluation. Each row reports the absolute Base or SFT* score. ***Avg.*** is the macro-average over the seven domains. Trained Judge values are paired percentile ranks rather than calibrated absolute scores. All values are rounded to two decimal places.

Evaluator	Model	Setting	Math	Code	Common.	Physical	Science	Writing	RealWorld	*Avg.*
Strong Judge	4B	Base	20.45	41.51	6.19	46.25	45.10	70.15	75.36	**43.57**
		SFT*	21.14	22.33	5.61	28.83	41.02	69.89	75.04	**37.69**
	9B	Base	22.49	46.71	7.40	45.63	44.70	73.95	80.24	**45.87**
		SFT*	24.67	23.97	7.35	30.78	44.16	71.47	80.18	**40.37**
	27B	Base	25.00	65.64	7.83	44.46	51.43	78.49	86.78	**51.38**
		SFT*	26.20	27.10	8.74	44.58	53.41	77.86	85.72	**46.23**
Trained Judge	4B	Base	48.83	52.96	50.22	50.37	52.19	53.64	53.00	**51.60**
		SFT*	51.17	47.04	49.78	49.63	47.81	46.36	47.00	**48.40**
	9B	Base	48.85	51.26	50.73	50.12	48.92	52.73	47.76	**50.05**
		SFT*	51.15	48.74	49.27	49.88	51.08	47.27	52.24	**49.95**
	27B	Base	48.96	50.23	51.15	48.32	50.12	51.81	39.77	**48.62**
		SFT*	51.04	49.77	48.85	51.68	49.88	48.19	60.23	**51.38**
Review Committee	4B	Base	4.85	23.55	2.06	42.36	57.18	62.65	76.88	**38.50**
		SFT*	5.16	10.38	1.76	17.63	54.12	62.32	76.72	**32.58**
	9B	Base	4.13	27.22	2.77	40.51	58.20	66.46	79.61	**39.84**
		SFT*	5.75	13.81	2.80	41.77	58.98	66.41	79.52	**38.43**
	27B	Base	5.52	47.11	4.00	35.85	63.71	69.68	81.77	**43.95**
		SFT*	6.59	20.36	4.03	23.94	61.64	69.33	81.69	**38.23**
Tool-Grounded Judge	4B	Base	5.72	0.00	9.47	42.43	43.18	61.15	69.42	**33.05**
		SFT*	9.05	0.00	9.18	18.11	40.90	61.23	69.12	**29.66**
	9B	Base	5.10	0.00	11.07	40.70	42.30	65.76	74.29	**34.17**
		SFT*	7.19	0.09	11.19	18.50	44.02	65.56	74.31	**31.55**
	27B	Base	6.92	0.00	11.14	36.04	50.05	70.03	77.26	**35.92**
		SFT*	8.06	0.00	11.12	23.92	53.93	68.82	76.74	**34.66**
Adversarial Tournament	4B	Base	5.81	24.40	9.01	31.76	35.48	50.05	66.87	**31.91**
		SFT*	8.93	25.26	8.62	32.33	32.76	49.41	66.68	**32.00**
	9B	Base	5.15	29.14	11.16	32.25	35.65	53.31	70.89	**33.93**
		SFT*	7.26	29.26	10.96	32.70	36.53	50.51	70.85	**34.01**
	27B	Base	7.10	50.97	11.79	30.94	41.27	57.93	75.78	**39.40**
		SFT*	8.24	50.79	13.55	31.12	42.87	56.68	75.63	**39.84**

Table 28: Detailed Novelty scores under instance-level evaluation. Each row reports the absolute Base or SFT* score. ***Avg.*** is the macro-average over the seven domains. Trained Judge values are paired percentile ranks rather than calibrated absolute scores. All values are rounded to two decimal places.

Evaluator	Model	Setting	Math	Code	Common.	Physical	Science	Writing	RealWorld	*Avg.*
Strong Judge	4B	Base	26.46	26.98	16.65	41.16	38.32	66.54	59.63	**39.39**
		SFT*	26.83	27.42	15.72	35.59	34.96	65.25	60.72	**38.07**
	9B	Base	29.90	30.10	21.06	45.04	40.10	72.02	63.57	**43.11**
		SFT*	28.61	29.26	19.40	38.63	38.88	67.95	64.97	**41.10**
	27B	Base	28.16	30.38	27.76	51.05	43.01	72.25	66.92	**45.65**
		SFT*	27.65	30.47	25.75	44.35	41.84	72.23	65.61	**43.99**
Trained Judge	4B	Base	51.59	52.08	50.05	49.56	50.02	52.72	55.91	**51.71**
		SFT*	48.41	47.92	49.95	50.44	49.98	47.28	44.09	**48.29**
	9B	Base	50.11	49.83	51.03	49.65	47.85	53.42	49.08	**50.14**
		SFT*	49.89	50.17	48.97	50.35	52.15	46.58	50.92	**49.86**
	27B	Base	49.50	49.25	48.26	49.42	51.25	51.21	41.54	**48.63**
		SFT*	50.50	50.75	51.74	50.58	48.75	48.79	58.46	**51.37**
Review Committee	4B	Base	35.39	28.50	0.74	38.57	52.54	66.97	75.01	**42.53**
		SFT*	34.37	27.30	0.55	36.23	47.06	66.22	74.63	**40.91**
	9B	Base	35.93	32.95	1.09	41.42	53.71	71.29	78.04	**44.92**
		SFT*	33.61	30.39	1.01	41.08	51.76	69.97	77.72	**43.65**
	27B	Base	33.47	32.87	0.96	46.54	56.81	71.67	79.68	**46.00**
		SFT*	33.01	30.77	0.81	45.37	53.84	70.26	79.51	**44.80**
Tool-Grounded Judge	4B	Base	31.24	25.26	4.92	41.02	43.33	49.49	62.96	**36.89**
		SFT*	24.43	25.32	4.52	40.02	41.98	48.20	62.52	**35.28**
	9B	Base	32.05	28.88	7.14	44.71	43.10	55.24	66.90	**39.72**
		SFT*	29.52	27.41	6.75	41.38	44.10	53.99	66.94	**38.58**
	27B	Base	28.77	26.77	8.68	49.47	49.49	54.31	70.41	**41.13**
		SFT*	29.13	26.36	8.42	49.48	50.72	53.95	70.11	**41.17**
Adversarial Tournament	4B	Base	35.19	30.58	7.29	38.84	34.82	60.81	65.94	**39.07**
		SFT*	27.33	30.25	7.07	38.46	30.68	59.94	65.47	**37.03**
	9B	Base	35.93	33.84	9.68	42.36	36.00	63.96	69.87	**41.66**
		SFT*	33.00	32.33	8.67	39.79	33.81	60.71	69.87	**39.74**
	27B	Base	32.83	32.46	7.80	46.88	39.65	62.59	71.38	**41.94**
		SFT*	33.20	32.39	9.12	47.41	37.25	61.70	70.45	**41.64**