TriCreativity: Benchmarking and Eliciting Combinational, Exploratory, and Transformational Creativity for Large Language Models

Anonymous authors
Paper under double-blind review

Abstract

Creativity is an important capability for large language models in scientific discovery, technological innovation, and open-ended problem solving. Existing benchmarks span multiple domains, but leave gaps in distinguishing the creative demands of tasks and assessing departures from existing conceptual frameworks. To this end, this paper introduces TriCreativity, a benchmark combining Boden's combinational, exploratory, and transformational taxonomy with multidimensional evaluation. We also introduce Counterfactual Transformation (CFT), a dataset that supplies counterfactual changes to fundamental premises. Models must develop additional representations and organizing principles to construct coherent and useful conceptual frameworks under the changed premises. Across 11 large language models spanning multiple families and sizes, creative strengths vary by creative demand and domain: even within exploratory creativity, mathematics, coding, and writing favor different models. Within-family scaling yields uneven gains. Most evaluated generic reasoning procedures improve selected tasks or metrics but struggle to achieve joint gains across creative objectives. Thus, we further introduce Taxonomy-Guided Creativity Elicitation (TCE), a training-free method that coordinates generation, search, selection, and verification around task-specific creative requirements and achieves the best aggregate performance among the compared procedures. Novelty and appropriateness are strongly associated in CFT but weakly related in constrained writing. Together with output-level cases, this suggests that when familiar premises fail, effective problem solving may depend not only on finding alternative answers but on constructing new representations and organizing principles.

1 Introduction

Creativity enables intelligent systems to form new connections, explore alternatives, and develop new representations when familiar approaches are insufficient (Boden, 1998). It is an important capability for large language models (LLMs) in scientific discovery, technological innovation, and open-ended problem solving (Ye et al., 2025; Lu et al., 2025). Systematic evaluation should reveal models' creative capabilities, their development, and opportunities for improvement.

Existing benchmarks evaluate creativity in mathematics, programming, and writing (Ye et al., 2025; Lu et al., 2025; Atmakuru et al., 2024); cross-domain suites examine output qualities and capability profiles (Hou et al., 2026; Beaty et al., 2026). Two challenges remain. First, domain-level comparisons provide less structure for distinguishing creative demands: combining familiar concepts, searching within a fixed solution space, or revising the space itself. Strengths can vary across demands and domains, so aggregates can obscure complementary capabilities. Boden's *combinational*, *exploratory*, and *transformational* creativity provide a theoretical framework for organizing these demands (Boden, 1998; 2004).

Second, evaluating framework-level transformation requires distinguishing conceptual departure from routine adaptation while assessing whether the resulting ideas remain coherent and useful.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

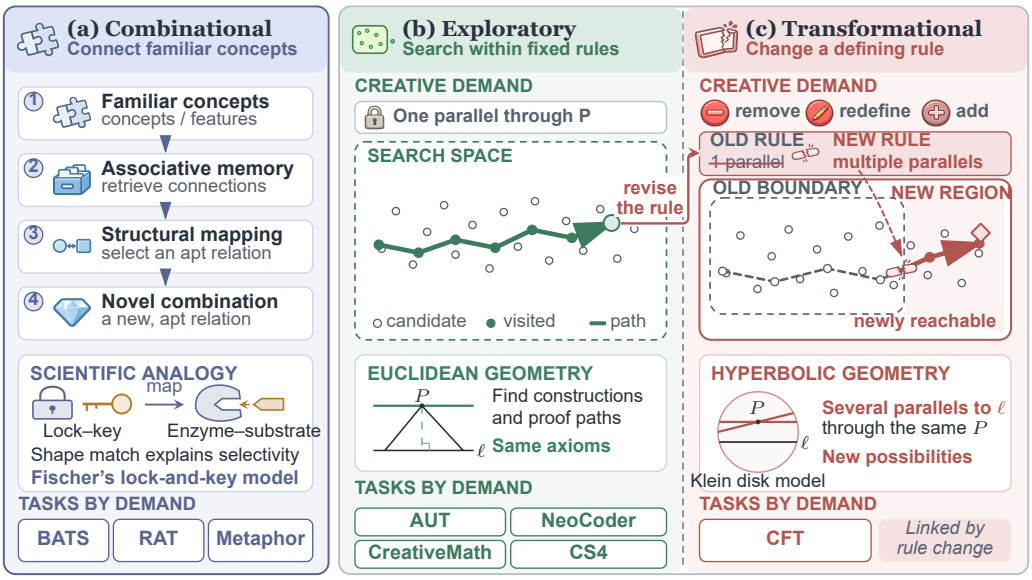


Figure 1: TriCreativity organized by dominant creative demands. (a) Combination maps lock–key fit to enzyme–substrate recognition (Fischer, 1902). (b) Exploration seeks constructions and proofs under fixed Euclidean axioms. (c) Transformation changes the parallel postulate to establish hyperbolic geometry and open new possibilities. The connector links exploration within a space to revision of its defining rules.

Transformational creativity changes the rules defining a conceptual space, enabling ideas the original framework excluded (Boden, 1998; 2004). New representations and organizing principles can make effective problem solving possible when familiar procedures no longer apply. Mathematical, coding, and constrained-writing benchmarks mainly assess alternatives within supplied task conditions (Ye et al., 2025; Lu et al., 2025; Atmakuru et al., 2024). Rule-altering reasoning procedures and narrative transformation broaden this scope (Suzuki & Banaei-Kashani, 2025; Schapiro et al., 2026), but assessing valuable framework-level innovation remains an open challenge.

To address these challenges, we introduce TriCreativity[1] (Figure 1). Following Boden, it organizes tasks by their dominant creative demands: combination connects familiar concepts; exploration searches within an established conceptual space; transformation changes its defining rules. Complementing this task organization, we evaluate outputs along fluency, flexibility, novelty, and appropriateness using task-supported native criteria. The taxonomy characterizes the kind of creativity a task primarily requires, whereas the dimensions characterize the qualities of its outputs.

To evaluate transformational creativity, we construct *Counterfactual Transformation* (CFT), a dataset of 1,308 items across six domains. Its counterfactual rule worlds invalidate premises supporting familiar solutions. Crucially, the benchmark supplies the premise change but not the resulting conceptual framework. Models develop additional representations, organizing principles, and mechanisms for coherent, useful ideas. Evaluating these model-added departures and their consequences makes framework-level innovation observable. Novelty and appropriateness are strongly associated in CFT but weakly related in constrained writing, consistent with new representations becoming especially useful when familiar premises no longer support effective solutions.

These distinctions also suggest a method-design principle: different creative demands may benefit from different test-time support. We therefore propose Taxonomy-Guided Creativity Elicitation (TCE), a training-free method that routes tasks to demand-specific operator pools and composes generation, search, selection, and verification. Generic procedures yield uneven gains across tasks

[1]Benchmark code, CFT data, and TCE implementation: `https://anonymous.4open.science/r/TriCreativity-5635/`.

and can improve one objective while weakening another; output-level analyses locate failures in candidate selection, constraint preservation, and completion. TCE achieves the strongest aggregate performance among the evaluated procedures, with task-specific trade-offs. This advantage persists under call-budget controls against generic sampling, revision, and tree search.

In sum, the contribution of this paper has three complementary aspects:

- **A Demand-Structured Creativity Benchmark.** TriCreativity organizes tasks with Boden's taxonomy while separately evaluating output quality through task-supported dimensions.
- **A Controlled Task for Framework-Level Transformation.** CFT evaluates model-added representations and organizing principles under changed governing premises that invalidate familiar solutions, distinguishing framework-level departures from routine adaptation.
- **Demand-Guided Elicitation and Empirical Insights.** TCE coordinates task-conditioned generation, search, selection, and verification. Across 11 models, strengths vary by creative demand and domain; within-family scaling improves tasks unevenly; and elicitation yields objective-specific gains and trade-offs.

The rest of this paper covers related work, the benchmark and method, experiments, discussion, and conclusions (Sections 2–6).

2 Related Work

Creativity and Psychological Foundations. Creativity combines originality with effectiveness or appropriateness (Runco & Jaeger, 2012; Amabile, 1982). Associative accounts emphasize connections among concepts, while divergent-thinking traditions examine idea production and variety (Mednick, 1962; Guilford, 1967). Boden distinguishes combination, exploration, and transformation of conceptual spaces (Boden, 1998; 2004; Wiggins, 2006). The taxonomy organizes task demands; fluency, flexibility, novelty, and appropriateness organize task-supported output qualities.

Creativity Benchmarks. CreativeMath, NeoCoder, and CS4 evaluate alternative mathematical solutions, constrained programs, and stories through domain-specific criteria (Ye et al., 2025; Lu et al., 2025; Atmakuru et al., 2024). CreativeBench operationalizes combination and exploration through executable code; TwistBench evaluates narrative reinterpretation through plot twists under coherence and realism requirements (Wang et al., 2026; Schapiro et al., 2026). Cross-domain suites broaden coverage through output-quality profiles (Hou et al., 2026; He et al., 2025; Beaty et al., 2026). TriCreativity adds demand-based organization across domains and CFT, which evaluates model-added representations and organizing principles under supplied counterfactual premises.

Test-Time Creativity Elicitation. Chain-of-Thought, Self-Consistency, Tree of Thoughts (ToT), and Self-Refine structure reasoning through decomposition, voting, search, and revision (Wei et al., 2022; Wang et al., 2023; Yao et al., 2023; Madaan et al., 2023). SELF-DISCOVER composes task-relevant reasoning modules, and Universe of Thoughts (UoT) develops Boden-inspired procedures including rule alteration (Zhou et al., 2024; Suzuki & Banaei-Kashani, 2025). TCE links creative demands and output objectives to specialized operators. Multidimensional outcomes, budget controls, and ablations assess the resulting workflows (Appendix A).

3 TriCreativity: Benchmark, Elicitation Method, and Evaluation

TriCreativity links task design, elicitation, and evaluation. We organize existing tasks by their dominant creative demands and construct CFT to assess framework-level transformation. TCE translates these demands into operator workflows. Task-native metrics then provide multidimensional profiles and normalized comparisons.

3.1 Benchmark Design and Construction

Task Taxonomy and Coverage. Eight tasks cover three primary demands (Table 1). Combinational tasks connect familiar concepts: the Bigger Analogy Test Set (BATS) tests relation comple-

Table 1: Tasks and native evaluators (cf. Appendix B.1). Flu., Flex., Nov., and App. denote fluency, flexibility, novelty, and appropriateness; supported mappings appear in Appendix B.2.

Task	Primary demand	Answer space	Native criteria	Reported dimensions
BATS	combinational	sparse, canonical	exact-match accuracy	App.
RAT	combinational	sparse, canonical	exact-match accuracy	App.
Metaphor	combinational	sparse, candidate set	exact-match accuracy	App.
AUT	exploratory	open, many-valid	use count, category coverage, originality, applicability	Flu./Flex. Nov./App.
NeoCoder	exploratory	open, executable	technique compliance, new-technique count, executable correctness	Flu./Nov. App.
CreativeMath	exploratory	open, multi-path	correctness, coarse/fine novelty	Nov./App.
CS4	exploratory	open, many-valid	constraint completion, lexical diversity, novelty, story quality	Flu./Flex. Nov./App.
CFT	transformational	open, many-valid	three-judge four-dimension rubric	Flu./Flex. Nov./App.

tion, the compound Remote Associates Test (RAT) tests remote association, and MUNCH-based Metaphor interprets metaphorical words through meaning-preserving lexical replacements (Gladkova et al., 2016; Bowden & Jung-Beeman, 2003; Tong et al., 2024). All use answer-set exact match. Exploratory tasks generate alternatives within specified conditions: the Alternate Uses Task (AUT) elicits uses (Wilson et al., 1960; Beaty & Johnson, 2021; Sun et al., 2024); CreativeMath requests correct solutions differing from supplied references; NeoCoder prohibits techniques while requiring executable correctness; CS4 requires constraint-controlled stories (Ye et al., 2025; Lu et al., 2025; Atmakuru et al., 2024). CFT changes the governing premises themselves. Native task specifications appear in Appendix B.1.

Counterfactual Transformation Data. CFT elicits conceptual-framework transformation by counterfactually invalidating premises supporting familiar solutions. Each item supplies an original system, changed governing premises, and functional goals, but not the resulting conceptual framework. Evaluation targets model-added representations and organizing principles for coherent, useful alternatives (Figure 2). System reconstruction exposes the operational consequences of these framework changes.

We combine 55 author-curated historical prototypes and 34 synthetic theme groups into 327 parent scenarios across six domains. GPT-5.4 generates and refines the scenarios; four nested profiles with 1, 2, 3, and 5 cumulative rule changes yield 1,308 items. Historical prototypes anchor interpretation; synthetic scenarios introduce counter-conventional rule worlds. Deterministic checks cover structure, rule and goal slots, metadata consistency, and exact duplicates, followed by a model-based answerability and difficulty pilot (Appendix B.3).

In Figure 2, mechanical work cannot compensate for insufficient source temperature. The response turns temperature grades into types organizing circuits, service accounting, and contracts. The rubric separates supplied changes and routine adaptation from model-added representations operating across system layers: novelty scores these departures; appropriateness scores coherent goal fulfillment. Case analysis and scoring anchors appear in Appendices D.4 and B.3.

3.2 Taxonomy-Guided Creativity Elicitation

TCE translates creative demands into prompted operators: relation building for combination, feasible alternative search for exploration, and rule propagation and framework construction for transformation (Figure 3). Routing, selection, and execution coordinate these operators on a fixed base model.

Routing, Selection, and Execution. *Task-level routing* selects a pool using creative demand, output form, and objective priorities. *Item-level selection* orders operators using the prompt and con-

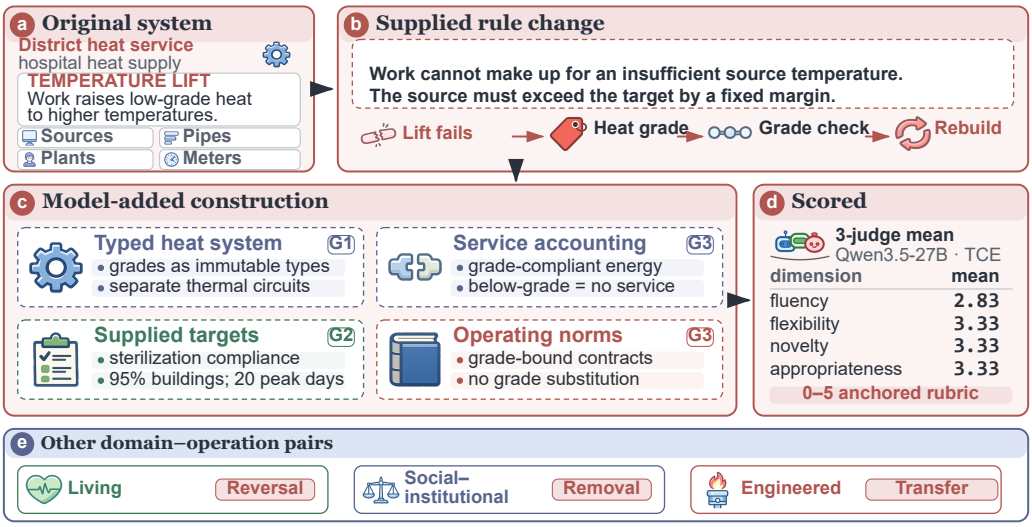


Figure 2: CFT in a counterfactual thermal world. A Qwen3.5-27B TCE response adds a *type-safe thermal compiler*, linking grade-segregated circuits, grade-compliant accounting, and contracts. The rule change and G1–G3 goals are supplied by the benchmark; scores average three judges. Case analysis and validity checks appear in Appendix D.4.

straints, covering high-priority objectives and placing extraction or mapping before generation and verification or selection.

Execution passes structured intermediate results, candidates, and checks between operators; task-specific finalization selects or synthesizes and formats a complete answer. Objective priorities guide workflow construction; evaluation uses equal-weight aggregation. TCE uses task-visible references and metadata, with held-out references and evaluator judgments confined to evaluation (Appendix F.2). Appendix F provides operator prompts and execution rules.

3.3 Evaluation Protocol

Native metrics retain their operational meanings within the supported dimensions. NeoCoder separates technique compliance, alternative-technique use, and executable correctness; CS4 separates constraint completion, lexical diversity, novelty, and story quality. A dimension can retain multiple fields, such as coarse and fine mathematical novelty (Appendix B.2).

We normalize native fields using fixed references within each comparison, average fields within dimensions, average supported dimensions within tasks, and give each of the eight tasks equal weight in Overall. Bounded fields use scale anchors; unbounded counts use empirical references (Appendix C.1.1). Main-experiment rubric scores average three cross-family judges and are checked against blinded human ratings, addressing presentation and family effects (Zheng et al., 2023; Wang et al., 2024; Liu et al., 2023). Scoring and reliability details appear in Appendices B.1 and C.3.

4 Experiments

We examine task and dimension profiles (**RQ1**); within-family scaling (**RQ2**); elicitation effectiveness and operator contributions (**RQ3**); and evaluation reliability and workflow robustness (**RQ4**).

4.1 Experimental Setup

We evaluate 11 models with direct prompting (Direct), answering the original task prompt (Table 2). The default generation temperature is 0.7; method-specific temperatures and output limits are detailed in Appendix F.1. Judge temperature is 0.2.

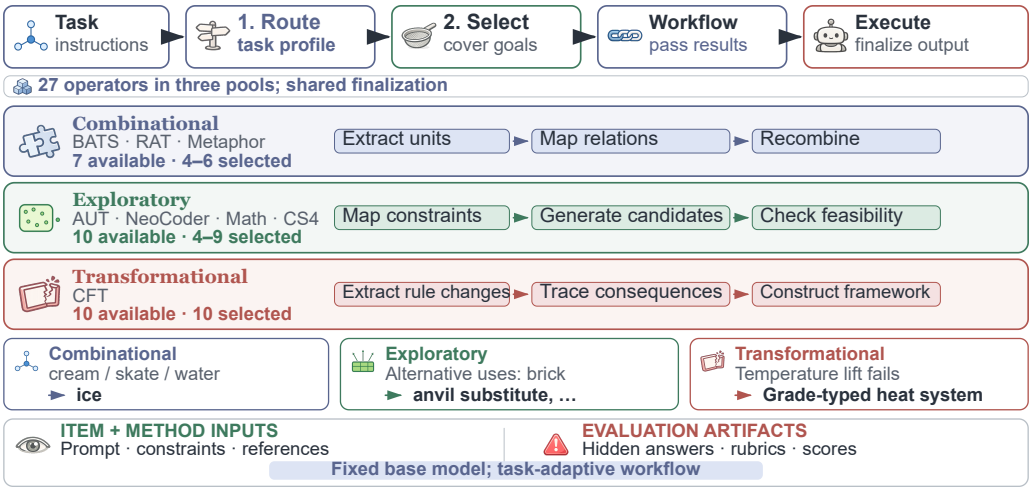


Figure 3: TCE routes creative demands to operator pools and selects objective-covering workflows. Intermediate results pass between steps before final synthesis and formatting. Available/selected denote pool size/operators chosen per workflow. Representative steps and examples are shown; details: Appendix F.

Elicitation fixes Qwen3.5-27B and evaluators, comparing Direct, Generic Creativity, Structured Chain-of-Thought (CoT), Self-Consistency (three-candidate exact-answer voting), Self-Refine, single-call ToT, and TCE. ToT compares three branches within one call, without external tree search. Ablations remove one TCE operator group at a time on matched one-fifth task subsets (Appendix C.4).

We compare elicitation methods at two levels. Table 3 evaluates complete procedures under their native execution structures. Table 4 allocates per-item call budgets derived from TCE's recorded workflow to generic sampling, iterative refinement, and tree search, testing whether additional calls enable these procedures to attain comparable creative performance.

Paired effects use 10,000 bootstrap resamples, clustering repeated versions by source problem, story, or parent scenario. Two-sided cluster-permutation tests receive Benjamini–Hochberg correction over 21 endpoints at target false discovery rate $q = 0.05$. Aggregation and supplementary ranks appear in Appendix C.2.

Benchmark evaluation code, CFT data and construction code, and implementations of TCE and the generic baselines are available in the anonymous repository.

4.2 Experimental Results

4.2.1 Task and Dimension Profiles Reveal Complementary Strengths (RQ1)

Task- and Domain-Specific Strengths. Aggregate strength does not identify the best model for every creative demand. Claude leads overall, while Gemini leads all three combinational tasks (Table 2). Even within exploration, writing favors GPT-5.2, mathematics Qwen3.5-27B, and coding Claude. Native dimensions further separate mathematical correctness from solution novelty, and technique compliance from executable correctness (Appendix C.1). These profiles make model selection specific to both task demands and desired output qualities.

Task Profiles and the Role of Novelty. Across models, mean Spearman correlation of task-composite ranks is 0.926 within the combinational block but 0.166 between exploratory tasks and CFT (Appendix Figure 10a), showing that task rankings convey different profiles. At the output level, median within-model Pearson correlation between novelty and appropriateness is 0.154 in CS4 and 0.861 in CFT; human CFT ratings also show a positive association ($r = 0.697$). Together with the inspected constructions, this supports the interpretation that new representations can help organize

Table 2: Direct-prompting scores of 11 models in $[0, 1]$ (higher is better). Task scores average normalized fields within dimensions, then dimensions equally; Overall averages eight tasks equally. **Bold**/underline: best/second best. Neo/Math: NeoCoder/CreativeMath. Scales: Appendix C.1.1.

	Combinational			Exploratory				Transf.	
Model	BATS	RAT	Metaphor	AUT	Neo	Math	CS4	CFT	Overall
Claude Opus 4.6	0.653	0.958	0.569	**0.974**	**0.786**	0.455	0.724	**0.674**	**0.724**
DeepSeek V3.2	0.609	0.771	0.499	0.796	0.660	0.397	0.648	0.550	0.616
Gemini 3.1 Pro	**0.664**	**0.986**	**0.588**	0.699	0.756	0.426	0.707	0.583	0.676
GPT-5.2	0.607	0.868	0.560	0.258	0.668	0.332	**0.754**	0.663	0.589
Kimi K2.5	0.622	0.958	0.528	0.519	0.700	0.355	0.667	0.656	0.626
Qwen3-Max	0.604	0.729	0.550	0.676	0.645	0.433	0.640	0.578	0.607
Qwen3.5-122B-A10B	0.574	0.424	0.490	0.419	0.553	0.471	0.622	0.627	0.523
Qwen3.5-27B	0.572	0.312	0.504	0.446	0.579	**0.474**	0.633	0.629	0.519
Qwen3.5-35B-A3B	0.546	0.222	0.480	0.365	0.559	0.433	0.591	0.613	0.476
Qwen3.5-4B	0.478	0.201	0.424	0.290	0.393	0.329	0.489	0.594	0.400
Qwen3.5-9B	0.523	0.215	0.447	0.375	0.478	0.375	0.550	0.610	0.447

Table 3: Qwen3.5-27B elicitation scores in $[0, 1]$ (higher is better). Task scores average dimensions; Overall averages eight tasks equally. **Bold**/underline: best/second best. Direct: direct prompting. AUT uses first-round references; Overall comparisons are within-table. Scales/native scores: Appendices C.1.1/C.2.

Method	BATS	RAT	Metaphor	AUT	Neo	Math	CS4	CFT	Overall
Direct	0.572	0.313	0.504	0.339	0.579	**0.472**	**0.639**	**0.628**	0.506
Generic Creativity	0.513	0.292	0.423	0.816	0.488	0.411	0.552	0.617	0.514
Structured CoT	0.611	0.618	0.510	0.509	0.392	0.403	0.482	0.468	0.499
Self-Consistency	**0.616**	**0.729**	**0.522**	0.468	0.401	0.404	0.492	0.467	0.512
Self-Refine	0.564	0.549	0.418	**0.915**	0.572	0.416	0.547	0.598	0.572
ToT (1 call)	0.587	0.493	0.483	0.824	0.474	0.403	0.594	0.562	0.552
TCE	0.588	0.521	0.482	0.867	**0.667**	0.459	0.617	0.626	**0.603**

useful solutions when familiar premises fail. Full metric relationships and scoring dependencies appear in Appendix D.1.

Cases Locate Task-Specific Strengths. GPT-5.2's constrained writing illustrates narrative organization and integration of multiple requirements, while Qwen3.5-27B's mathematical outputs show structured derivation and attention to problem conditions. These observable features locate strengths beyond final scores. Detailed Qwen–Claude comparisons, including reverse cases where Claude is stronger, and writing cases appear in Appendix D.2.

4.2.2 Uneven Creativity Gains from Within-Family Scaling (RQ2)

Within-family scaling yields different gains across tasks and output qualities. Across dense Qwen3.5 models from 4B to 27B, NeoCoder correctness rises from 0.097 to 0.420 and CS4 full-constraint completion from 0.092 to 0.308. Normalized task composites rise from 0.393 to 0.579 on NeoCoder and from 0.594 to 0.629 on CFT (Table 2). Native metrics and dense and mixture-of-experts (MoE) profiles appear in Appendix C.5.8.

4.2.3 Effects of Elicitation on Creativity (RQ3)

Evaluated Generic Procedures Show Objective Trade-offs. Self-Consistency raises RAT accuracy from 0.313 to 0.729, and Self-Refine increases AUT first-round retained uses from 9.01 to 39.55. On NeoCoder, Self-Consistency improves prohibited-technique compliance while lowering correctness and new-technique use. Generic Creativity raises CS4 novelty at the expense of constraint completion and lexical flexibility; all five generic procedures reduce CreativeMath correctness relative to Direct (Appendix D.3).

Joint Gains and Task-Dependent Trade-offs. TCE achieves the highest task-averaged normalized score, 0.603, compared with 0.572 for the strongest generic baseline, Self-Refine (Table 3). It leads the NeoCoder task composite and the CFT novelty and appropriateness dimensions, combining gains across complementary creative objectives. Dimensional analyses identify joint improvements over direct prompting across all four AUT fields and in NeoCoder originality and correctness. On CFT, the highest novelty and appropriateness scores accompany task-specific trade-offs in fluency and flexibility. Paired effects and tests appear in Appendix C.2.

Table 4: Call-budget controls on common items (normalized scores; higher is better). Overall averages eight tasks equally. **Bold**/underline: best/second best. Neo/Math: NeoCoder/CreativeMath. Counts, native scores, and protocol: Appendix C.5.9.

Method	BATS	RAT	Metaphor	AUT	Neo	Math	CS4	CFT	Overall
TCE	**0.588**	0.521	0.482	**0.868**	**0.667**	**0.665**	0.599	**0.667**	**0.632**
Best-of-N	0.586	0.354	**0.516**	0.489	0.521	0.593	**0.610**	0.605	0.534
Self-Refine	0.503	**0.535**	0.353	0.440	0.623	0.568	0.530	0.632	0.523
Multi-call ToT	0.575	0.278	0.495	0.588	0.651	0.560	0.597	0.642	0.548

Controlling for Additional Inference Calls. Increasing the call allowance of generic sampling, revision, and search does not recover TCE's aggregate performance in this comparison. TCE scores 0.632 overall, versus 0.548 for the strongest control, multi-call ToT (Table 4). Its advantages include alternative-technique use in NeoCoder, mathematical novelty, and CFT flexibility and novelty, supporting the value of task-conditioned coordination alongside additional generation and refinement.

All four methods use common items and the same task-specific scoring protocol. Call allocation, judge panels, and sample counts appear in Appendix C.5.9.

Table 5: Operator-group ablations on matched one-fifth subsets. Full: three-run mean at temperature 0.7; w/o: relative score change after removal (negative means lower). Units/Relations: unit extraction/relation abstraction; F/A: fluency/appropriateness. Full results and groups: Appendix C.4.

Combinational					
Task	Metric	Full	w/o Units	w/o Relations	
RAT	Accuracy	0.782	−33.8%	−29.4%	
Exploratory					
Task	Metric	Full	w/o Exploration	w/o Novelty	w/o Selection
NeoCoder	Compliance (F)	0.586	+42.9%	+42.9%	+32.9%
NeoCoder	Originality	0.225	−72.0%	−77.6%	−81.4%
NeoCoder	Correctness (A)	0.517	−51.5%	−49.1%	−56.3%
CreativeMath	Correctness	0.882	+2.2%	+2.2%	+2.2%
CreativeMath	Coarse novelty	0.568	−34.5%	−40.3%	−30.1%
Transformational					
Task	Metric	Full	w/o Rule extraction	w/o Breakage	w/o Reconstruction
CFT	Fluency	2.823	−0.3%	+0.1%	+0.4%
CFT	Novelty	3.140	−11.6%	−11.2%	−10.6%
CFT	Appropriateness	3.286	−3.9%	−3.3%	−3.1%

Contributions of Creativity-Specific Operators. Operator ablations separate output production from creative quality (Table 5). Removing unit extraction or relation abstraction lowers RAT accuracy. On CFT, removing rule-extraction, breakage, or reconstruction groups barely changes fluency but reduces novelty by 10.6–11.6%. On NeoCoder, higher technique compliance coincides with lower originality and executable correctness; CreativeMath similarly trades slightly higher correctness for lower novelty. Similar production or compliance scores can therefore conceal losses in useful alternative construction. Group definitions and the full matrix appear in Appendix C.4.

Cases Explain Positive and Negative Effects. Inspected traces locate three points where additional reasoning can lose value: candidate selection, constraint preservation, and solution completion. Exact-answer voting aggregates shared RAT solutions but selects the first candidate when all three open-ended outputs differ. Self-Refine expands AUT lists with both new uses and repeated constructions. Other verified cases show repeated boundary-condition omissions, unfinished calculations, or reversed semantic roles in CS4. These patterns connect dimensional trade-offs to task-appropriate selection and validity checks (Appendix D.3).

On inspected CFT items, preliminary analysis in Structured CoT and Self-Consistency leaves rule implementation and validation unfinished within the response limit, while a separate Self-Refine revision completes more of the proposal. Response allocation thus also affects whether reasoning reaches a usable construction (Appendix D.3).

4.2.4 Evaluation Reliability and Workflow Robustness (RQ4)

We report consistency intraclass correlation, ICC(C,3), for three-rater mean scores. The LLM panel has macro ICC(C,3) $= 0.792$. Three annotators blindly score 140 Direct outputs (40 CreativeMath, 40 CS4, 60 CFT), yielding macro human ICC(C,3) $= 0.866$. Human–LLM Pearson correlations exceed 0.80 for estimable CreativeMath novelty and CS4 metrics; CFT reaches 0.547–0.637, with human ICC(C,3) 0.742–0.812 and LLM-panel ICC(C,3) 0.699–0.764. Full reliability results appear in Appendix C.3.

On the 60 human-scored CFT outputs, human novelty–appropriateness correlation is 0.697 (parent-cluster bootstrap 95% interval $[0.549, 0.816]$), remaining 0.566 after log-length control. Full-sample cross-judge correlations also remain positive but are lower than same-judge correlations (summarized as 0.355 versus 0.813). The association therefore extends across human and automated ratings, with strength sensitive to the scoring arrangement (Appendix C.5.4). Judge removal retains novelty-rank Spearman correlation ≥ 0.982; panel correlation is stable under log-length control ($0.861 \rightarrow 0.857$).

CFT mean scores favor TCE in novelty and appropriateness on both Qwen3.5-27B and DeepSeek V3.2 (Appendices C.5 and C.2).

5 Discussion

TriCreativity supports improvement at three levels: task-demand profiles locate model strengths, native dimensions identify output qualities needing improvement, and response traces reveal actionable failures in selection, constraint preservation, and completion. TCE turns these requirements into demand-conditioned elicitation; budget controls and ablations examine the benefits of its coordination and operators.

CFT examines valuable ideas beyond familiar frameworks. Supplied premise changes create conditions for transformation; models develop additional representations and organizing principles. Separately evaluating these departures and their operational consequences distinguishes useful framework innovation from routine adaptation.

6 Conclusion and Limitations

TriCreativity combines Boden-based task-demand organization, multidimensional evaluation, and CFT for assessing framework innovation. Results reveal complementary strengths and uneven development. TCE yields joint gains on selected objectives and the strongest aggregate performance in the evaluated comparisons. Together, the benchmark, dataset, and elicitation method connect creativity assessment with targeted improvement.

Limitations. Task categories co-vary with domain and evaluator, and CFT is the only task in its category. Public tasks carry exposure risk (Yang et al., 2023; Golchin & Surdeanu, 2025); synthetic scenarios may retain generator patterns. Human agreement is assessed on a limited subset, and judge families overlap with evaluated families. Scaling covers one family; cross-base elicitation covers CFT. Detailed data, evaluator, and statistical qualifications appear in Appendices B.3 and C.3.

Reproducibility Statement

The anonymous code supplement contains the CFT dataset, construction code and prompts, benchmark adapters, scoring templates, elicitation workflows, and offline tests. Its `README.md` provides entry points, and `docs/DATA.md` documents task-data preparation. Appendices B.1, B.3, F.3, and C.3 describe the task, construction, workflow, and evaluation protocols.

AI Use Statement

Generative AI tools assisted with literature discovery, synthetic dataset generation and refinement, experimental design and implementation, result analysis and interpretation, and manuscript drafting, translation, and language polishing. The authors take responsibility for the final content and research results.

References

Teresa M. Amabile. Social psychology of creativity: A consensual assessment technique. *Journal of Personality and Social Psychology*, 43(5):997–1013, 1982. doi: 10.1037/0022-3514.43.5.997.

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, 2024. URL https://arxiv.org/abs/2410.04197.

Roger Beaty, Vijeta Deshpande, Clin K. Y. Lai, Anna Attuch, Namrata Shivagunde, Swastik Roy, Rajkumar Pujari, Paul V. DiStefano, Sherin Muckatira, Claire E. Stevenson, Mikhail Gronas, and Anna Rumshisky. AGC-Bench: Measuring artificial general creativity, 2026. URL https://arxiv.org/abs/2607.01152.

Roger E. Beaty and Dan R. Johnson. Automating creativity assessment with SemDis: An open platform for computing semantic distance. *Behavior Research Methods*, 53:757–780, 2021. doi: 10.3758/s13428-020-01453-w.

Margaret A. Boden. Creativity and artificial intelligence. *Artificial Intelligence*, 103(1–2):347–356, 1998. doi: 10.1016/S0004-3702(98)00055-1.

Margaret A. Boden. *The Creative Mind: Myths and Mechanisms*. Routledge, London, 2 edition, 2004. ISBN 9780415314527.

Edward M. Bowden and Mark Jung-Beeman. Normative data for 144 compound remote associate problems. *Behavior Research Methods, Instruments, & Computers*, 35(4):634–639, 2003. doi: 10.3758/BF03195543.

Emil Fischer. Syntheses in the purine and sugar group. Nobel Lecture, December 12, 1902. URL https://www.nobelprize.org/uploads/2018/06/fischer-lecture-2.pdf.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. Analogy-based detection of morphological and semantic relations with word embeddings: What works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pp. 8–15. Association for Computational Linguistics, 2016. doi: 10.18653/v1/N16-2002.

Shahriar Golchin and Mihai Surdeanu. Data contamination quiz: A tool to detect and estimate contamination in large language models. *Transactions of the Association for Computational Linguistics*, 13:809–830, 2025. doi: 10.1162/tacl.a.20.

J. P. Guilford. *The Nature of Human Intelligence*. McGraw-Hill, New York, 1967.

Zicong He, Boxuan Zhang, Weihao Liu, Ruixiang Tang, and Lu Cheng. What shapes a creative machine mind? comprehensively benchmarking creativity in foundation models, 2025. URL https://arxiv.org/abs/2510.04009.

Zhaoyi Joey Hou, Bowei 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*, 2026. URL https://arxiv.org/abs/2510.20091.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2511–2522. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.153.

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, Albuquerque, New Mexico, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.naacl-long.141.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Sarnoff A. Mednick. The associative basis of the creative process. *Psychological Review*, 69(3): 220–232, 1962. doi: 10.1037/h0048850.

Sarnoff A. Mednick and Martha T. Mednick. *Remote Associates Test: Examiner's Manual*. Houghton Mifflin, Boston, 1967.

Vaishnavh Nagarajan, Chen Henry Wu, Charles Ding, and Aditi Raghunathan. Roll the dice & look before you leap: Going beyond the creative limits of next-token prediction. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 45395–45436. PMLR, 2025. URL https://proceedings.mlr.press/v267/nagarajan25a.html.

Mark A. Runco and Garrett J. Jaeger. The standard definition of creativity. *Creativity Research Journal*, 24(1):92–96, 2012. doi: 10.1080/10400419.2012.650092.

Samuel Schapiro, Alexi Gladstone, Heng Ji, and Roger E. Beaty. TwistBench: Benchmarking transformational creativity in LLMs via literary plot twists, 2026. URL https://schapiro.ai/files/TwistBench.pdf. Contributed oral presentation at the COLM 2026 Workshop on Scientific Understanding of Foundation Models.

Luning Sun, Hongyi Gu, Rebecca Myers, and Zheng Yuan. A new dataset and method for creativity assessment using the alternate uses task. In *Intelligent Computers, Algorithms, and Applications: Third BenchCouncil International Symposium, IC 2023, Sanya, China, December 3–6, 2023, Revised Selected Papers*, volume 2036 of *Communications in Computer and Information Science*, pp. 125–138. Springer Singapore, 2024. doi: 10.1007/978-981-97-0065-3_9.

Yuto Suzuki and Farnoush Banaei-Kashani. Universe of thoughts: Enabling creative reasoning with large language models, 2025. URL https://arxiv.org/abs/2511.20471.

Xiaoyu Tong, Rochelle Choenni, Martha Lewis, and Ekaterina Shutova. Metaphor understanding challenge dataset for LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3517–3536, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.193.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Proceedings*

of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 9440–9450, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.511.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

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, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1546.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Geraint A. Wiggins. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-Based Systems*, 19(7):449–458, 2006. doi: 10.1016/j.knosys.2006.04.009.

R. C. Wilson, P. R. Christensen, P. R. Merrifield, and J. P. Guilford. *Alternate Uses, Form A: Manual of Administration, Scoring, and Interpretation*. Sheridan Supply Company, Beverly Hills, California, 2nd preliminary edition, 1960.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. Rethinking benchmark and contamination for language models with rephrased samples, 2023. URL https://arxiv.org/abs/2311.04850.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract-Conference.html.

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. doi: 10.1609/aaai.v39i24.34760.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

Pei Zhou, Jay Pujara, Xiang Ren, Xinyun Chen, Heng-Tze Cheng, Quoc V. Le, Ed H. Chi, Denny Zhou, Swaroop Mishra, and Huaixiu Steven Zheng. SELF-DISCOVER: Large language models self-compose reasoning structures. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e41efb03e20ca3c231940a3c6917ef6f-Abstract-Conference.html.

Appendix Guide

The appendices begin with expanded related work, followed by benchmark specifications, empirical evidence, and implementation references.

Appendix	Contents	Page
A	Expanded related work	13
B	Task specifications, dataset construction, and scoring anchors	14
C	Complete results, evaluator validation, and robustness analyses	21
D	Metric relationships, model cases, and elicitation case analysis	35
E	Historically grounded CFT examples	42
F	Workflow inputs, operators, and execution	44
G	Judge, routing, and baseline prompt templates	45
H	Operator index, execution template, and representative instructions	54

A Expanded Related Work

A.1 Creativity and Psychological Foundations

Creativity combines originality with effectiveness; associative and divergent-thinking accounts motivate relation building and the production of alternatives (Runco & Jaeger, 2012; Amabile, 1982; Mednick, 1962; Guilford, 1967). Boden distinguishes combining familiar concepts, exploring structured conceptual spaces, and changing their defining rules (Boden, 1998; 2004; Wiggins, 2006). We use these distinctions to organize dominant task demands; individual tasks can draw on multiple creative abilities.

A.2 Creativity Benchmarks

Universe of Thoughts (UoT) constructs combinational, exploratory, and transformational creative-reasoning procedures and evaluates them on three open-ended design problems (Suzuki & Banaei-Kashani, 2025). Its transformational procedure surfaces explicit and hidden rules, alters high-impact rules, and searches the resulting spaces; the evaluated problems retain their stated constraints, placing the taxonomy at the procedure level. CreativeBench builds combinational and exploratory code challenges with executable scoring and pairs them with the inference-time steering method EvoRePE (Wang et al., 2026). Going Beyond Next-Token studies combinational and exploratory limits in controlled algorithmic settings (Nagarajan et al., 2025). TwistBench operationalizes literary transformation through plot twists scored for surprise, coherence, plausibility, and diversity against expert-authored work (Schapiro et al., 2026). These designs cover procedure-level rule alteration, executable creativity, controlled search, and narrative reinterpretation; TriCreativity adds controlled tasks for examining whether models can break with existing conceptual frameworks, open up new possibilities, and develop valuable ideas.

CreativityPrism combines cross-domain tasks with quality, novelty, and diversity measures (Hou et al., 2026); AGC-Bench examines domain-specific strengths and a common creativity factor (Beaty et al., 2026). TriCreativity integrates cross-domain multidimensional evaluation with demand-based task organization, controlled defining-rule changes, and output-level diagnosis of model and elicitation profiles.

From Capability Goals to Task Design. UoT identifies and alters explicit or hidden rules through a reasoning procedure, then searches the modified spaces. CFT uses counterfactual rule worlds as evaluation settings, whose governing principles can depart from ordinary real-world laws. Each item supplies changed premises but leaves the resulting conceptual framework to the model. Models develop additional representations, organizing principles, and operating procedures, explaining how their ideas coherently meet the stipulated goals. The dataset covers six domains and 1,308 items, with four dimensions evaluating the resulting constructions. The rubric distinguishes model-added framework departures from routine adaptation to the supplied changes.

Table 6: Complementary designs in closely related creativity evaluations. The comparison distinguishes the evaluated setting, the role of creative-demand organization, and the evidence used to assess outputs.

Work	Evaluated setting	Creativity organization	Evaluation focus
UoT	Three open-ended design problems	Boden categories structure reasoning procedures	Creative value and feasibility of proposals
CreativeBench	Executable coding challenges	Combinational and exploratory task construction	Code-based creative problem solving
TwistBench	Narrative reinterpretation through plot twists	Transformational narrative setting	Surprise, coherence, realism, and diversity
CreativityPrism	Cross-domain creative tasks	Output-quality profiles	Quality, novelty, and diversity
AGC-Bench	Cross-domain creativity assessment	Domain and shared capability structure	Domain-specific strengths and a common factor
TriCreativity	Cross-domain tasks and counterfactual rule worlds	Dominant task demands and demand-guided elicitation	Native output dimensions and model-added framework departures

Modular Reasoning and Creativity Elicitation. SELF-DISCOVER selects atomic reasoning modules and composes explicit task-level reasoning structures (Zhou et al., 2024); UoT organizes creative procedures around Boden's taxonomy. Building on these modular approaches, TCE is a taxonomy-guided elicitation method that composes creativity-specific operators and passes intermediate artifacts between them to support task-conditioned improvement.

B Benchmark Construction and Metrics

B.1 Per-Task Specifications

This subsection fixes the unit of evaluation and executable benchmark score for every task. The reported n counts adapter-expanded evaluation items; one source record may generate several evaluation prompts.

BATS ($n = 4{,}000$). We use a bidirectional 4,000-item semantic sample from the Bigger Analogy Test Set (Gladkova et al., 2016): 200 items from each of its 10 lexicographic and 10 encyclopedic relation categories, excluding its morphological categories. The adapter registers `bats_accuracy`; after lowercasing and trimming the parsed `target` field, an item scores one when it is a member of the record's `target_words`, and zero otherwise. The report is the arithmetic mean of these indicators.

```
Complete the word analogy:

berlin : germany :: conakry : ?

Relation type: country - capital

Reference target_words: ["guinea"]
[excerpt: JSON answer-format instructions omitted]
```

RAT ($n = 144$). We use all 144 compound remote associate problems introduced and normed by Bowden and Jung-Beeman (Bowden & Jung-Beeman, 2003), a compound-form descendant of the Remote Associates Test (Mednick, 1962; Mednick & Mednick, 1967). The parsed `word` is stripped and lowercased; `rat_accuracy` is one when it exactly equals the likewise normalized single reference word, and zero otherwise. The report is its mean.

```
Find a single word that connects with all three words to form common
compound words or phrases.

Words: cottage/swiss/cake

Reference answer: cheese
[excerpt: worked "cheese" example and final instruction omitted]
```

Metaphor: MUNCH Metaphor Interpretation ($n = 2{,}953$). We use all 2,953 metaphorical sentences in MUNCH's paraphrase-generation set (Tong et al., 2024). Each contains a highlighted word marked as metaphorically used and asks for a one-word target-domain paraphrase that preserves the sentence meaning. Our local evaluator uses deterministic candidate-set exact match against the retained crowdsourced apt paraphrases: after parsing, lowercasing, and trimming the single `word`, `metaphor_accuracy` is one for membership in the record's `candidate_answers` and zero otherwise; the report is the mean.

```
Paraphrase the given sentence by substituting the highlighted word
with another word. The substitution should be a single word.

Sentence: Latest corporate unbundler reveals laid-back *approach*:
Roland Franklin, who is leading a 697m pound break-up bid for DRG,
talks to Frank Kane

Replace the highlighted word (*approach*) with a single word that
maintains the same meaning of the sentence.

Reference candidate_answers:
["direction", "method", "plan", "view", "attitude"]
```

AUT ($n = 73$). We use an author-constructed 73-prompt AUT-style evaluation that asks for alternative uses of cue nouns. The format follows the Alternate Uses tradition (Wilson et al., 1960; Guilford, 1967); the local stimulus and reference-response pools combine heterogeneous SemDis verbal-creativity materials and Cambridge AUT data (Beaty & Johnson, 2021; Sun et al., 2024). Because some source materials were originally collected for verb-generation and word-association studies rather than AUT itself, this is not a published standardized 73-item AUT instrument. Its head score, `aut_fluency`, is the cumulative number of uses retained after parsing, the implementation's LLM-judge validity/deduplication stage (including its documented fallback), and MiniLM-based semantic deduplication. Elaboration is the mean non-stopword length per use. Flexibility is $(C + 3O)/\sqrt{U}$, where C is the number of covered reference clusters, O the number of out-of-cluster uses, and U the number of embedded uses. Originality sums the normalized nearest-cluster excess above 1.5 and is aggregated as the mean of the top 10% of item scores. The adapter additionally reports `aut_applicability` as the number of judge-retained valid uses divided by the raw-use count; all other displayed AUT fields are arithmetic means over items.

```
What are some creative uses for a [sic] artist? List as many creative
and
alternative uses as you can think of for a artist. Each use should be
a brief phrase or sentence.
```

NeoCoder ($n = 1{,}194$ Expanded Items). The source contains 199 Codeforces problems; the adapter creates one item for each paired problem statement and prohibited-technique list (Lu et al., 2025). Correctness is one when the extracted Python program runs within the timeout and its output matches every annotated expected line (using the NeoCoder type-agnostic comparator when available). `follow_constraints` is one when the detected-technique and prohibited sets have an empty intersection; `new_techniques` counts detected techniques absent from the human-solution technique set, and `new_techniques_ratio` divides that count by all detected techniques. The aliases are exact: `fluency=mean(follow_constraints)`, `originality=mean(new_techniques)`, and `appropriateness=mean(correctness)`.

```
A. Line Trip Programming constraints: DO NOT use the following
techniques
- for loop There is a road, which can be represented as a number line.
You are located in the point $$$0$$$ of the number line, and you want
to travel from the point $$$0$$$ to the point $$$x$$$, and back to the
point $$$0$$$. [excerpt]
```

CreativeMath ($n = 605$ Expanded Items). For each source problem, let n_{ref} be its total reference-solution count and k the number shown in an item. The 400 source problems are expanded over visible prefixes of length $k = 1, \ldots, n_{\text{ref}}$ (Ye et al., 2025). Each item has three binary decisions: correctness (all three judges must vote YES), coarse novelty against solutions $1{:}k$ (majority YES), and fine novelty against the held-out solutions $k+1{:}n_{\text{ref}}$ (majority YES, evaluated only after a correct, coarsely novel answer and when $k < n_{\text{ref}}$). The benchmark ratios divide the respective positive counts by all items; the two conditional ratios are coarse-novel over correct and fine-novel over coarse-novel. The derived aliases are $\texttt{originality} = 0.7\,\texttt{novelty_ratio} + 0.3\,\texttt{novel_unknown_ratio}$ and `appropriateness=correctness_ratio`.

```
One half of the water is poured out of a full container. Then one
third of the remainder is poured out. Continue the process: one fourth
of the remainder for the third pouring, one fifth of the remainder for
the fourth pouring, etc. After how many pourings does exactly one
tenth of the original water remain?

Reference solution 1: 1)Model the amount left in the container as
follows:
After the first pour $\frac12$ remains, after the second $\frac12
\times \frac23$ remains, etc. This becomes the product $\frac12 \times
\frac23 \times \frac34 \times \cdots \times \frac{9}{10}$. Note that
the terms cancel out leaving $\frac{1}{10}$. [excerpt]
```

CS4 ($n = 250$). CS4 comprises 50 base stories at each of 7, 15, 23, 31, and 39 constraints (Atmakuru et al., 2024). Per item, fluency is the binary indicator that the mean three-judge satisfied-constraint count reaches the stated total; grammar, coherence, likability, relevance, and novelty are judge means on 1–5 scales. Flexibility is the product of unique/total 2-, 3-, and 4-gram ratios. Appropriateness is $(\textit{relevance}/5)(\textit{coherence}/5)$ and QUC is $(\textit{coherence}/5) \times \textit{fluency}$. Overall fields first average constraint levels within each story and then average the 50 story means; RCS is mean QUC at 39 constraints minus mean QUC at 7 constraints.

```
Story Instruction: Write a story in less than 500 words about an
introverted protagonist with uncontrollable intuition, who experiences
others' emotions intensely, as she navigates a challenging encounter
during the holiday season.

SelectedConstraints:
1. The protagonist suffers physical discomfort when overwhelmed by
emotions (nausea, shaking, etc.).
2. The protagonist is challenged by the need to engage in public
spaces.
3. The unknown man realizes that the protagonist can feel his
emotions.
[excerpt: constraints 4--7 and BaseStory omitted]
```

Counterfactual Transformation (CFT; $n = 1{,}308$). The new task uses the 327-parent construction described below, assembled from 55 historical prototypes and 34 synthetic theme groups. Each judge rates fluency, flexibility, novelty, and appropriateness on a 0–5 scale in 0.5 increments under the v10 rubrics. Each item's dimension scores are the arithmetic means of the three ratings. Full definitions and anchors are in Appendix B.3. Historically grounded non-Euclidean, relativity, and quantum examples are reproduced in Appendix E.

B.2 Interpretive Dimension Mapping

Task-specific report fields are assigned to interpretive dimensions while preserving each task's scale. Table 7 records the fields used for the four dimensions. Diagnostic fields outside this mapping include AUT elaboration and CS4 grammar.

The common display range supports within-task comparison, not a common cross-task measurement scale. CS4 flexibility measures lexical diversity, AUT flexibility measures category coverage, and NeoCoder's fluency-labeled field measures prohibited-technique compliance. Native fields remain necessary for interpreting composites and elicitation gains.

Table 7: Task-specific report fields and their dimension assignments. Only supported dimensions are listed. The sparse-answer combinational tasks map normalized exact-match accuracy to appropriateness; the remaining mappings preserve each task's report-field semantics.

Task	Supported dimension mapping
BATS	A = `mean_bats_accuracy`
RAT	A = `mean_rat_accuracy`
Metaphor	A = `mean_metaphor_accuracy`
AUT	F = `mean_aut_fluency`; Fx = `mean_aut_flexibility`; N = `mean_aut_originality`; A = `mean_aut_applicability`
NeoCoder	F = `mean(follow_constraints)`; N = `mean(new_techniques)`; A = `mean(correctness)`
CreativeMath	N = `novelty_ratio` (coarse) and `novel_unknown_ratio` (fine); A = `correctness_ratio`
CS4	F = `fluency`; Fx = product diversity; N = `novelty`; A = normalized relevance $\times$ normalized coherence
CFT	F = `fluency`; Fx = `flexibility`; N = `novelty`; A = `appropriateness`

NeoCoder Constraint Compliance. The native report field `fluency` equals `mean(follow_constraints)`: an item scores one when detected and prohibited technique sets do not intersect. This is a constraint-compliance indicator, not a count of valid ideas; a compliant program can still fail execution. We retain the source field and its organizational mapping for reproducibility, while interpreting compliance, new-technique count, and correctness separately.

B.3 CFT Data Construction and Scoring Rubric

B.3.1 Construction Counts

The dataset combines 55 historical prototypes with 34 synthetic theme groups. The historical route contributes one parent per prototype; the synthetic route contributes eight parents per theme group, yielding 272. The resulting 327 parents each produce four nested active-rule profiles containing 1, 2, 3, and 5 rules, respectively, with 327 records in each profile.

Data Provenance and Pilot Validation. The authors assembled the historical prototype pool from broadly documented scientific, technological, biological, representational, and institutional rule shifts and organized each prototype into the benchmark schema. The synthetic route draws from an author-defined taxonomy of six domains and 34 subdomain themes. GPT-5.4 generated the parent scenarios and performed one refinement pass under the `quality_v2` profile. Deterministic checks validated structure, taxonomy fields, rule and goal slots, metadata consistency, and exact duplicates.

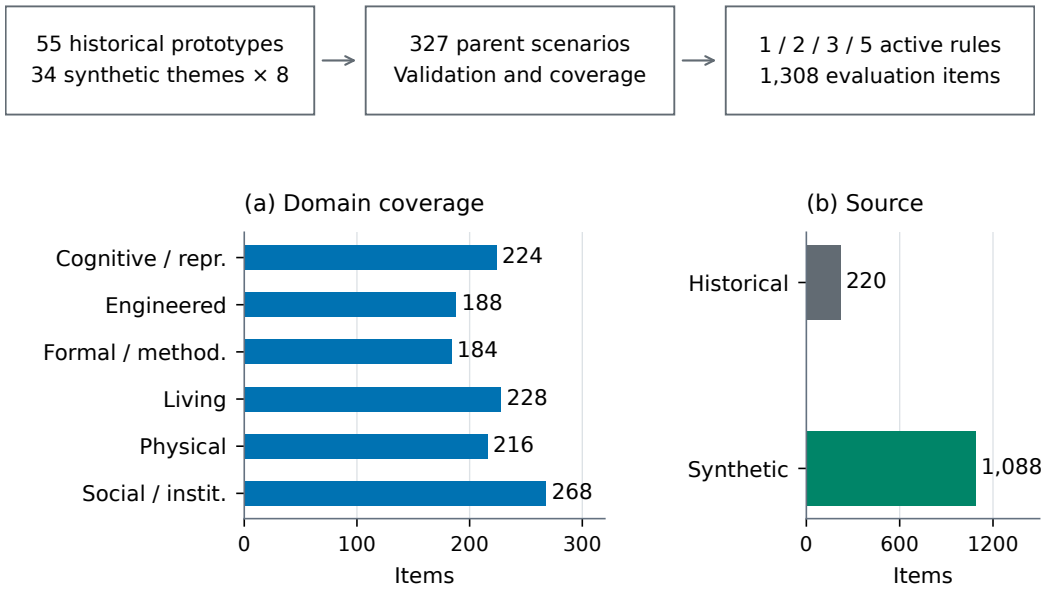


Figure 4: Construction and composition of the CFT dataset: 55 historical prototypes and 34 synthetic theme groups yield 327 parent scenarios and 1,308 nested-rule items across six domains and six rule-operation types.

Construction Prompt Resources. The anonymous code supplement stores the generation and refinement templates in `evalscope/dataprocess/transformation/prompt_catalog.json`. Its `historical_quality_v2` and `random_quality_v2` entries specify the generation profiles; `historical_refinement_system` and `random_refinement_system` specify refinement. The same directory contains `generate_transformation_dataset.py`, `generate_random_synthetic_dataset.py`, and the construction catalog.

As a post-construction pilot, DeepSeek, Qwen2.5-7B, Qwen2.5-72B, and Qwen3-8B answered all 1,308 items. A separately configured DeepSeek judge assessed task completion, goal coverage, rule compatibility, novelty, and appropriateness. This pilot characterized answerability and difficulty; item admission followed successful generation and deterministic validation.

Table 8: CFT item counts by domain.

Domain	Items
Cognitive–representational	224
Engineered	188
Formal–methodological	184
Living	228
Physical	216
Social–institutional	268
Total	1,308

B.3.2 Four-Dimensional Rubric

The rubric uses half-point anchors. Fluency counts non-redundant operational mechanism families; flexibility measures integrated switching across conceptual categories; novelty measures representational departure beyond the supplied rule change and an expected reconstruction; and appropriateness measures the conditional validity and operational quality of the candidate framework under the rewritten rules and stated goals. Completion diagnostics and dimension scores remain separate.

Table 9: CFT active-rule occurrences by operation type. Counts are occurrences across items, so an item with multiple active rules contributes to multiple occurrences.

Rule operation	Active-rule occurrences
Amplification	256
Conditionalization	1,314
Domain Transfer	304
Removal	279
Reversal	714
Suppression	730
Total	3,597

Fluency Anchors.

0 No usable ideational output: empty, refusal-like, severely off-task, meaningless error text, or no reconstructive solution attempt.

0.5 One fragment, slogan, or isolated suggestion without a working mechanism family.

1 Very low fluency: one non-generic mechanism family, or several labels that are not operationally distinct.

1.5 Low fluency: two weakly distinct mechanism families, with much of the answer made of generic implementation chores.

2 Modest fluency: two or three concrete mechanism families, while most details are routine governance, monitoring, rollout, audit, training, fallback, or recordkeeping around the same idea.

2.5 Moderate fluency: three or four non-redundant mechanism families with some concrete execution details; the normal range for a complete but ordinary comprehensive answer.

3 Good fluency: at least five distinct operational mechanism families, most not reducible to generic compliance, monitoring, or rollout.

3.5 Strong fluency: many non-redundant mechanism families across the rebuilt system, with clear causal roles and little padding; uncommon.

4 Very strong fluency: a dense set of distinct mechanism families creates multiple ways for the rule world to operate, validate itself, and recover from failure; rare.

4.5 Exceptional fluency: unusually rich ideational production with many independent, concretely operationalized, mutually non-duplicative mechanism families.

5 Near-ceiling fluency: extraordinary non-redundant ideational quantity comparable to a publishable system design space, not merely a long implementation plan.

Flexibility Anchors.

0 No usable breadth: empty, refusal-like, severely off-task, meaningless error text, or no reconstructive solution attempt.

0.5 One narrow fragment or perspective with no category shift.

1 Very narrow flexibility: remains inside one subsystem, stakeholder view, representation, time horizon, or mechanism family.

1.5 Low flexibility: a second category is present but shallow or subordinate to the same operating idea.

2 Limited flexibility: several sections are covered, but mainly as routine expansions of one mechanism family or implementation template.

2.5 Moderate flexibility: several genuinely different categories or system layers are used, but their connections are partial; the normal range for a complete but ordinary comprehensive answer.

3 Good flexibility: coherent movement across categories such as representation, records, interfaces, validation logic, incentives, institutions, failure semantics, and time horizons, with shifts that affect design decisions.

3.5 Strong flexibility: multiple categories interact nontrivially, changing perspective across stakeholders, system layers, evidence standards, control loops, and failure modes rather than listing them separately; uncommon.

4 Very strong flexibility: broad, integrated category switching reorganizes how the system is represented, operated, measured, governed, and repaired; rare.

4.5 Exceptional flexibility: unusually wide and coherent conceptual breadth with cross-category dependencies not obtainable by expanding a standard operating template.

5 Near-ceiling flexibility: extraordinary integrated breadth comparable to a new design grammar spanning representations, institutions, operations, evidence, failure semantics, and temporal adaptation.

Novelty Anchors.

0 No meaningful reconstruction: empty, refusal-like, severely off-task, meaningless error text, or no reconstructive solution attempt.

0.5 Fragments or slogans that gesture at change but do not specify a working mechanism.

1 Generic repair under the old or obvious frame: standard rollout, training, audit, monitoring, governance, automation, dashboarding, coordination, or checklist compliance without a new system model.

1.5 Competent expected reconstruction: concrete implementation of the rewritten rules, but mainly operationalizing the premise supplied by the item.

2 Strong expected reconstruction: integrated mechanisms, sequencing, failure handling, migration, or validation across modules, while leaving the old object model, measurement frame, causal explanation, and validity logic intact; the normal ceiling for a good non-paradigmatic answer.

2.5 Local representational departure beyond the supplied change: an additional state variable, validity condition, measurement object, causal explanation, unit of action, responsibility logic, or coordination grammar constrains at least two design parts.

3 Minimal novel framework: an unstated representational primitive, invariant, validity condition, measurement object, causal explanation, unit of action, responsibility logic, or coordination grammar organizes at least two major system layers beyond an expected reconstruction.

3.5 Strong novel framework: the additional frame organizes at least three major system layers and changes what operators record, interfaces expose, decisions count as valid, or failures mean.

4 Hidden-premise replacement: a false premise of the old system is replaced by a system-wide validity standard that changes what counts as evidence, state, responsibility, success, or legitimate action.

4.5 Field-level transformation: a reusable reasoning frame and family of practices could generalize beyond the item while remaining operationally grounded.

5 Axiom-level transformation: a new reasoning space comparable in depth to major scientific or mathematical frame shifts; exceptional.

Appropriateness Anchors.

0 Unworkable, empty, refusal-like, severely off-task, meaningless error text, or no reconstructive solution attempt.

0.5 Barely relevant fragments with no reliable execution path.

1 Superficially relevant but generic, vague, authority-dependent, missing enforceable rule–goal traceability, or only minimally compatible with the rewritten rule world.

1.5 Partially relevant plan with isolated mechanisms but major gaps in goal coverage, rule compatibility, integration, validation, migration, enforcement, failure handling, or legacy fit.

2 Locally runnable reconstruction: several concrete mechanisms are compatible with the supplied rules, but goal coverage, cross-module integration, validation, or failure handling remains incomplete.

2.5 Coherent candidate framework: most active rules and goals have concrete execution paths, and at least two design elements are traceably derived from the proposed framework, with some integration gaps remaining.

3 Good conditional fit: all active rules and stated goals are addressed through connected mechanisms, and the interfaces, records, and validation procedures align with the proposed framework.

3.5 Strong operational fit: the reconstruction is coherent across at least three major system layers, with measurable validation, explicit failure semantics, and workable fallback behavior.

4 Operationally complete reconstruction: mechanisms, interfaces, records, institutions, terminology, validation, migration, and fallback are mutually consistent and traceable to the supplied rules and goals.

4.5 Robust system-replacement quality: the reconstruction remains coherent under normal and failure conditions, with few hidden dependencies, clear migration and audit paths, and explicit trade-offs.

5 Exceptional rule-world fit: a self-consistent, comprehensive, verifiable, and failure-aware operating framework capable of replacing the old system; exceptional execution quality.

C Complete Results and Evaluation Reliability

This appendix reports the complete results, reliability estimates, ablations, stability analyses, and weighting sensitivity underlying Section 4.2.

C.1 Complete Direct Results

C.1.1 Normalization, Task Composites, and Overall Scores

Let e index an evaluated model–procedure configuration. For task T, let $M_{e,T,j}$ be the native score for report field j, aggregated according to the task's rules under configuration e, and let $U_{T,j} > 0$ be its fixed reference divisor. The normalized field $z_{e,T,j}$ and task composite $S_{e,T}$ are

$$z_{e,T,j} = \frac{M_{e,T,j}}{U_{T,j}}, \qquad S_{e,T} = \frac{1}{|\mathcal{D}_T|} \sum_{d \in \mathcal{D}_T} \frac{1}{|J_{T,d}|} \sum_{j \in J_{T,d}} z_{e,T,j}. \tag{1}$$

Here, $\mathcal{D}_T$ is the set of dimensions supported by task T, d indexes one such dimension, $J_{T,d}$ is the nonempty set of native field indices mapped to dimension d, and vertical bars denote set cardinality. Thus, fields are averaged within each dimension before dimensions are averaged equally. The model comparisons in Table 2 use fixed divisors and retain the zero origin. Native ratios use $U_{T,j} = 1$; all CFT dimensions and CS4 novelty use $U_{T,j} = 5$. CS4 retains the evaluator's zero-valued non-story outcome alongside its ordinary 1–5 novelty anchors. Count or accumulation fields without a calibrated full-score anchor use the largest aggregate among the fixed 11 Direct models in Table 2; these empirical references are fixed across the evaluated models.

The overall normalized score O_e uses equal task weights:

$$O_e = \frac{1}{|\mathcal{T}|} \sum_{T \in \mathcal{T}} S_{e,T}, \qquad |\mathcal{T}| = 8. \tag{2}$$

Here, $\mathcal{T}$ is the set of eight benchmark tasks, $S_{e,T}$ is the task composite defined above for configuration e, and $|\mathcal{T}|$ is the number of tasks. Each task contributes weight $1/8$. The task-internal dimension weights and field normalization remain unchanged.

Table 7 specifies the field mappings. BATS, RAT, and Metaphor support appropriateness only; AUT, CS4, and CFT support four dimensions, and NeoCoder three. For a given configuration, write A, N_c, and N_f for its normalized CreativeMath correctness, coarse-novelty, and fine-novelty fields,

Table 10: Empirical reference divisors for the model comparisons in Table 2.

Task	Native field	Divisor
AUT	Fluency / retained uses	155.9041
AUT	Flexibility	37.1296
AUT	Originality	194.4930
NeoCoder	New-technique count	0.2806

respectively. Its CreativeMath composite S_{Math} equally weights the two supported dimensions and averages both fields within novelty: $S_{\text{Math}} = \frac{1}{2}A + \frac{1}{4}N_c + \frac{1}{4}N_f$. Unsupported dimensions are omitted, derived aliases are not counted twice, and composites are not stretched again. The components and composites used in Tables 2 and 3 lie within $[0, 1]$. Empirical divisors are fixed references; scores exceeding them yield normalized values above 1, retained without clipping.

Native aggregation remains task-specific: the model comparison retains cumulative AUT rounds, whereas the method experiments use first-round AUT. AUT originality keeps its top-10% aggregation and CS4 its within-story then across-story averaging. Table 3 applies the same normalization and aggregation rules to the method comparison's native scores. Its three AUT divisors are the maxima across the fixed seven first-round methods: 39.547945 (fluency), 21.785497 (flexibility), and 75.040583 (originality). NeoCoder retains 0.2806; bounded fields retain their scale anchors. Divisors are fixed across methods, with no minimum subtraction or clipping. The model and method tables retain their original reporting cohorts and AUT round definitions, so their overall scores support within-table comparisons, not direct cross-table comparisons. Supplementary ranks are computed separately for each native field and are invariant to positive rescaling.

Reference-Scale Sensitivity. We recalibrate the four empirical divisors after leaving each model out of the reference pool, and multiply all four by 0.5 or 2, for 13 settings. The compared entities remain fixed and sensitivity values above one are not clipped. Every task leader in Table 2 is unchanged. Minimum AUT/NeoCoder rank correlations across models are $0.982/0.955$; other tasks do not depend on these empirical divisors. Separately multiplying all four empirical divisors by 0.5 or 2 preserves Claude as the overall model leader and TCE as the overall method leader; some lower positions change. Supplementary mean ranks in Table 16 are invariant to positive field rescaling.

Table 11: CreativeMath Direct native components; all three ratios use the full 605 expanded items. The composite uses the hierarchical normalization above.

Model	Appropriateness	Coarse novelty	Fine novelty	Composite
Claude Opus 4.6	0.6512	0.4182	0.1008	0.4554
DeepSeek V3.2	0.5851	0.3339	0.0826	0.3967
Gemini 3.1 Pro	0.5471	0.4975	0.1107	0.4256
GPT-5.2	0.5355	0.2149	0.0413	0.3318
Kimi K2.5	0.5223	0.3174	0.0595	0.3554
Qwen3-Max	0.6479	0.3603	0.0744	0.4326
Qwen3.5-122B-A10B	0.7008	0.4083	0.0760	0.4715
Qwen3.5-27B	0.7041	0.4033	0.0860	0.4744
Qwen3.5-35B-A3B	0.6496	0.3603	0.0744	0.4335
Qwen3.5-4B	0.5405	0.1917	0.0413	0.3285
Qwen3.5-9B	0.6050	0.2446	0.0463	0.3752

Figure 10a shows the full Spearman rank-correlation matrix of task composites across the 11 Direct model instances. Block averages over unique task pairs are 0.926 within the combinational block, 0.645 combinational–exploratory, 0.417 within the exploratory block, 0.215 combinational–CFT, and 0.166 exploratory–CFT.

CreativeMath correctness is effectively tied at the top: Qwen3.5-27B scores 0.7041 (426/605) and Qwen3.5-122B-A10B scores 0.7008 (424/605). Their paired difference is 0.0033 with exact McNemar $p = 0.917$.

Table 12: Item-level Pearson correlation between novelty and appropriateness, computed *within* each Direct model to remove between-model level differences, using each task's mapped fields (Appendix B.2). CM-c and CM-f are CreativeMath coarse and fine novelty against correctness. CFT is the only task where every model exceeds $+0.8$; NeoCoder is negative for 6 of 11 models.

Model	CFT	CM-c	AUT	CM-f	CS4	NeoCoder
Claude Opus 4.6	$+0.939$	$+0.620$	$+0.172$	$+0.245$	$+0.146$	-0.011
Gemini 3.1 Pro	$+0.861$	$+0.905$	$+0.189$	$+0.321$	$+0.109$	-0.028
GPT-5.2	$+0.911$	$+0.487$	$+0.581$	$+0.193$	$+0.203$	-0.082
Kimi K2.5	$+0.876$	$+0.652$	$+0.126$	$+0.241$	$+0.673$	-0.033
Qwen3-Max	$+0.873$	$+0.553$	$+0.119$	$+0.209$	$+0.186$	-0.007
DeepSeek V3.2	$+0.873$	$+0.596$	$+0.274$	$+0.253$	$+0.176$	-0.124
Qwen3.5-122B-A10B	$+0.854$	$+0.543$	$+0.424$	$+0.187$	$+0.144$	$+0.098$
Qwen3.5-35B-A3B	$+0.830$	$+0.551$	$+0.444$	$+0.208$	$+0.154$	$+0.099$
Qwen3.5-27B	$+0.832$	$+0.533$	$+0.285$	$+0.199$	$+0.052$	$+0.043$
Qwen3.5-9B	$+0.836$	$+0.460$	$+0.304$	$+0.178$	$+0.138$	$+0.148$
Qwen3.5-4B	$+0.811$	$+0.449$	$+0.252$	$+0.191$	$+0.302$	$+0.020$
Median	$+0.861$	$+0.551$	$+0.274$	$+0.208$	$+0.154$	-0.007
Items per model	1,304–1,308	605	73	605	250	1,194

Table 13: Leaders on selected native Direct diagnostic fields; n is the number of evaluated items. These component-level leaders need not coincide with the task-composite leaders in the main table.

Task	Native diagnostic field	Field leader	Score	n
BATS	Analogy accuracy	Gemini 3.1 Pro	0.6637	4,000
RAT	Remote-association accuracy	Gemini 3.1 Pro	0.9861	144
Metaphor	Candidate-set exact match	Gemini 3.1 Pro	0.5875	2,953
AUT	Retained-use count (fluency)	Claude Opus 4.6	155.9041	73
NeoCoder	Correctness	Claude Opus 4.6	0.7169	1,194
CreativeMath	Correctness ratio	Qwen3.5-27B	0.7041	605
CS4	Full-constraint completion (fluency)	GPT-5.2	0.5480	250
CFT	Fluency	Claude Opus 4.6	3.3215	1,308

Table 14 retains selected native diagnostic fields for inspecting absolute performance; the main-table task composites instead include all supported dimensions using the rule above.

Table 14: Selected native Direct diagnostic fields for the fixed model set. AUT is a retained-use count; CFT is on its task-specific rubric scale; the other columns retain their task-specific report units.

Model	BATS	RAT	Metaphor	AUT	NeoCoder	CreativeMath	CS4	CFT
Claude Opus 4.6	0.6535	0.9583	0.5686	155.9041	0.7169	0.6512	0.460	3.3215
DeepSeek V3.2	0.6090	0.7708	0.4992	117.4795	0.6801	0.5851	0.244	2.7538
Gemini 3.1 Pro	0.6637	0.9861	0.5875	88.6164	0.6608	0.5471	0.452	2.8087
GPT-5.2	0.6068	0.8681	0.5598	26.3699	0.4648	0.5355	0.548	3.2407
Kimi K2.5	0.6218	0.9583	0.5283	58.4384	0.6223	0.5223	0.284	3.2076
Qwen3-Max	0.6038	0.7292	0.5496	83.9726	0.6390	0.6479	0.188	2.7983
Qwen3.5-122B-A10B	0.5737	0.4236	0.4903	39.6164	0.4481	0.7008	0.292	3.0022
Qwen3.5-27B	0.5723	0.3125	0.5042	45.6438	0.4204	0.7041	0.308	3.0206
Qwen3.5-35B-A3B	0.5457	0.2222	0.4802	36.6712	0.3283	0.6496	0.232	2.9541
Qwen3.5-4B	0.4785	0.2014	0.4236	17.5205	0.0972	0.5405	0.092	2.8558
Qwen3.5-9B	0.5232	0.2153	0.4470	30.0000	0.2245	0.6050	0.172	2.9332

Table 15 uses the display normalizations in Appendix C.1.1. The fluency summary averages AUT, NeoCoder, CS4, and CFT fluency fields; flexibility averages AUT, CS4, and CFT flexibility fields. These equal-weight summaries are descriptive indices, not validated measures of a common fluency or flexibility construct. Native task components remain the basis for diagnosis.

Table 15: Descriptive averages of normalized fields grouped under fluency and flexibility; not a cross-task latent-ability scale. CFT contributes 1,308 scored items for nine models, 1,306 for Qwen3.5-4B, and 1,304 for Qwen3.5-122B-A10B.

Model	Fluency	Flexibility
Claude Opus 4.6	0.725	0.881
Gemini 3.1 Pro	0.601	0.768
GPT-5.2	0.476	0.592
Kimi K2.5	0.500	0.738
Qwen3.5-27B	0.462	0.695
Qwen3.5-122B-A10B	0.435	0.683
Qwen3.5-35B-A3B	0.422	0.664
Qwen3-Max	0.472	0.762
Qwen3.5-9B	0.406	0.661
DeepSeek V3.2	0.479	0.792
Qwen3.5-4B	0.335	0.620

C.2 Complete Method-by-Task–Metric Matrix

Table 16 supplements the normalized-score comparison in Table 3 with mean ranks, first-place counts, and novelty-specific ranks.

Table 16: Complete rank summaries for elicitation methods on Qwen3.5-27B. Mean rank averages all 21 task–metric ranks; first-place count counts top-ranked task–metric combinations; novelty mean rank averages six novelty/originality fields. Arrows indicate better performance. **Bold**: best; underline: second best. Task–metric scores and ranks appear in Table 19.

Method	Mean Rank↓	First-Place Count↑	Novelty Mean Rank↓
Direct	3.48	<u>5</u>	3.67
Generic Creativity	4.33	1	3.67
Structured CoT	5.00	1	6.50
Self-Consistency	4.76	4	5.33
Self-Refine	<u>3.43</u>	3	<u>3.00</u>
ToT (1 call)	4.05	0	3.33
TCE	**2.95**	**7**	**2.50**

Paired Inference Settings. All 21 task–metric effects enter one exploratory testing family. We use 10,000 percentile-bootstrap resamples (seed 20260916), preserving the task's aggregation. NeoCoder and CreativeMath resample source problems, CS4 resamples stories, and CFT resamples parent scenarios; other tasks resample items. Each sampled cluster carries all retained versions and both methods together. Two-sided method-label permutation tests independently swap TCE and Direct within each source cluster, using 10,000 permutations and the add-one Monte Carlo correction. Benjamini–Hochberg adjustment covers all 21 resulting p-values at target false discovery rate $q = 0.05$. We report BH-adjusted p-values.

Aggregation. AUT originality is the mean of the top 10% of item scores, while CS4 overall fields first average constraint levels within each story and then average story-level means. All methods, including Direct, are evaluated on first-round AUT outputs. An AUT response with no retained uses receives applicability 0 under the scoring rule.

Clustering Units. CreativeMath clusters use the complete problem text with whitespace normalized; identical question texts share a cluster. NeoCoder uses source problems, CS4 uses stories, and CFT uses parent scenarios. These source units keep repeated versions together during resampling.

Each cell in Table 19 is *task-scale mean (within-row rank)*. Higher is better for every included item. Scores are displayed to three decimals; ranks are computed before rounding. Ranks use descending average-rank tie handling: values tied over ranks $a, \ldots, b$ each receive $(a + b)/2$.

Table 17: All 21 TCE-minus-Direct paired effects. Intervals use 10,000 cluster-bootstrap replicates; p_{BH} adjusts two-sided cluster-permutation tests jointly over all 21 endpoints. Native metric scales are retained.

Task	Metric	Δ	95% interval	p_{BH}
RAT	Accuracy	$+0.208$	$[+0.125, +0.292]$	< 0.001
Metaphor	Accuracy	-0.022	$[-0.039, -0.006]$	0.012
BATS	Accuracy	$+0.016$	$[+0.005, +0.026]$	0.006
AUT	Fluency	$+25.822$	$[+22.753, +28.849]$	< 0.001
AUT	Flexibility	$+13.545$	$[+12.329, +14.730]$	< 0.001
AUT	Originality	$+53.920$	$[+42.863, +62.007]$	< 0.001
AUT	Applicability	$+0.122$	$[+0.043, +0.201]$	0.006
NeoCoder	Constraint compliance	-0.012	$[-0.044, +0.021]$	0.585
NeoCoder	Originality	$+0.059$	$[+0.026, +0.095]$	0.001
NeoCoder	Appropriateness	$+0.064$	$[+0.031, +0.097]$	0.001
CreativeMath	Correctness	-0.026	$[-0.060, +0.007]$	0.216
CreativeMath	Coarse novelty	-0.009	$[-0.051, +0.033]$	0.785
CreativeMath	Fine novelty	$+0.010$	$[-0.015, +0.035]$	0.585
CS4	Fluency	-0.100	$[-0.185, -0.011]$	0.042
CS4	Flexibility	$+0.020$	$[+0.016, +0.025]$	< 0.001
CS4	Novelty	-0.004	$[-0.106, +0.097]$	0.942
CS4	Appropriateness	-0.008	$[-0.032, +0.015]$	0.585
CFT	Novelty	$+0.243$	$[+0.211, +0.275]$	< 0.001
CFT	Appropriateness	$+0.039$	$[+0.012, +0.066]$	0.008
CFT	Flexibility	-0.115	$[-0.136, -0.094]$	< 0.001
CFT	Fluency	-0.216	$[-0.237, -0.195]$	< 0.001

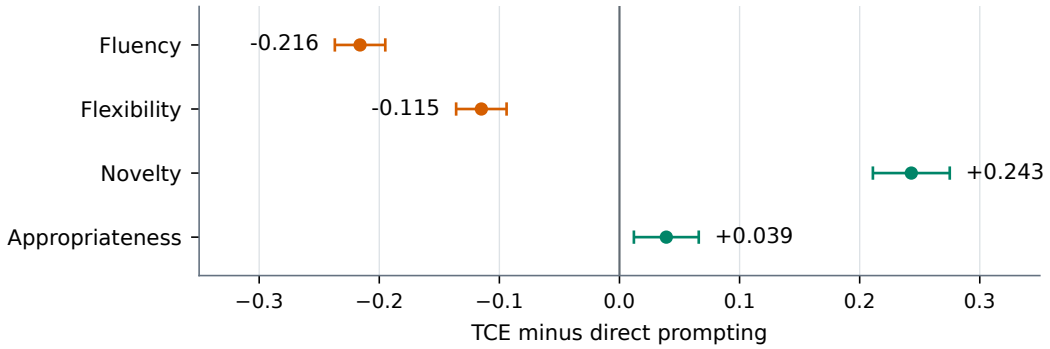


Figure 5: Paired CFT effects of TCE relative to direct prompting. Points show differences and horizontal bars show 95% parent-scenario cluster-bootstrap intervals from Table 17, in native metric units.

Table 18: CFT three-method comparison on two base families. Entries are mean dimension scores; Direct denotes direct prompting. Higher is better; bold marks the best method within each row.

Dimension	Base model	Direct	Generic	TCE
Novelty	Qwen3.5-27B	2.974	3.037	**3.217**
	DeepSeek V3.2	2.474	2.782	**3.256**
Appropriateness	Qwen3.5-27B	3.294	3.279	**3.333**
	DeepSeek V3.2	2.804	3.083	**3.353**
Flexibility	Qwen3.5-27B	**3.278**	3.169	3.162
	DeepSeek V3.2	2.965	3.026	**3.234**
Fluency	Qwen3.5-27B	**3.023**	2.846	2.806
	DeepSeek V3.2	2.754	2.723	**2.896**

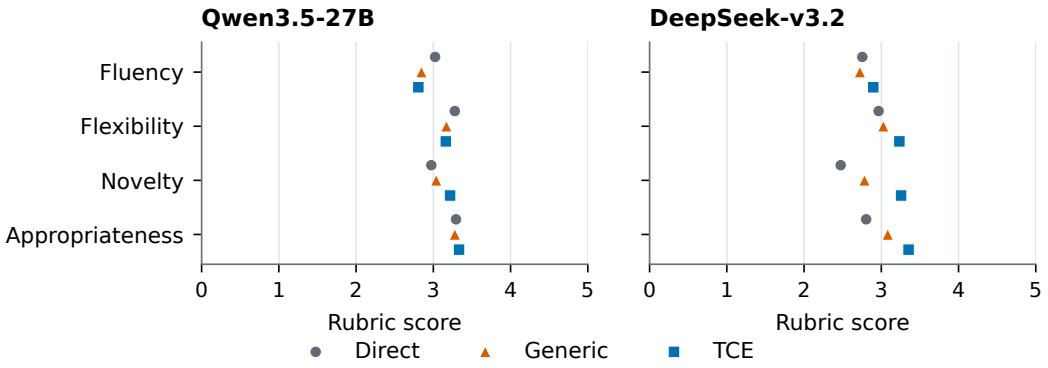


Figure 6: CFT on two base families, shown on the full native 0–5 score range. Points show mean dimension scores. TCE has higher novelty and appropriateness on both bases; the other dimensions vary by base.

Table 19: Complete task-scale-mean and rank matrix for seven methods and 21 task–metric combinations on Qwen3.5-27B. Generic denotes the generic creativity instruction, SC denotes Structured CoT, Cons. denotes Self-Consistency, Refine denotes Self-Refine, and ToT denotes the single-call ToT-style prompt.

Task	Metric	TCE	Generic	SC	Cons.	Refine	ToT	Direct
RAT	accuracy	0.521 (4)	0.292 (7)	0.618 (2)	0.729 (1)	0.549 (3)	0.493 (5)	0.313 (6)
Metaphor	fit	0.482 (5)	0.423 (6)	0.510 (2)	0.522 (1)	0.418 (7)	0.483 (4)	0.504 (3)
BATS	accuracy	0.588 (3)	0.513 (7)	0.611 (2)	0.616 (1)	0.564 (6)	0.587 (4)	0.572 (5)
AUT	fluency	34.836 (2)	31.315 (4)	14.945 (5)	12.452 (6)	39.548 (1)	32.890 (3)	9.014 (7)
AUT	flexibility	21.785 (1)	18.268 (3)	7.552 (6)	6.272 (7)	19.056 (2)	18.006 (4)	8.241 (5)
AUT	originality	67.181 (3)	70.469 (2)	32.864 (5)	30.904 (6)	75.041 (1)	60.988 (4)	13.261 (7)
AUT	applicability	0.694 (6)	0.695 (5)	0.874 (1)	0.857 (2)	0.784 (4)	0.825 (3)	0.571 (7)
NeoCoder	fluency	0.633 (6)	0.632 (7)	0.896 (2)	0.903 (1)	0.694 (4)	0.777 (3)	0.645 (5)
NeoCoder	originality	0.248 (1)	0.133 (4)	0.043 (7)	0.049 (6)	0.157 (3)	0.095 (5)	0.188 (2)
NeoCoder	appropriateness	0.484 (1)	0.358 (4)	0.126 (6)	0.123 (7)	0.463 (2)	0.306 (5)	0.420 (3)
CreativeMath	correctness	0.671 (2)	0.620 (3)	0.601 (5)	0.594 (6)	0.603 (4)	0.558 (7)	0.697 (1)
CreativeMath	coarse novelty	0.397 (3)	0.326 (7)	0.338 (6)	0.347 (5)	0.364 (4)	0.400 (2)	0.406 (1)
CreativeMath	fine novelty	0.099 (1)	0.076 (6)	0.071 (7)	0.080 (5)	0.092 (3)	0.095 (2)	0.088 (4)
CS4	fluency	0.230 (4)	0.160 (7)	0.215 (5)	0.253 (3)	0.315 (2)	0.200 (6)	0.330 (1)
CS4	flexibility	0.950 (1)	0.668 (4)	0.581 (6)	0.587 (5)	0.517 (7)	0.840 (3)	0.930 (2)
CS4	novelty	2.574 (6)	3.132 (1)	2.516 (7)	2.585 (4)	2.782 (3)	2.807 (2)	2.578 (5)
CS4	appropriateness	0.771 (4)	0.755 (5)	0.627 (6)	0.610 (7)	0.798 (1)	0.775 (3)	0.779 (2)
CFT	novelty	3.217 (1)	3.037 (2)	2.408 (7)	2.413 (6)	2.870 (4)	2.744 (5)	2.974 (3)
CFT	appropriateness	3.333 (1)	3.279 (3)	2.211 (6)	2.209 (7)	3.189 (4)	2.977 (5)	3.294 (2)
CFT	flexibility	3.162 (3)	3.169 (2)	2.467 (6)	2.456 (7)	3.100 (4)	2.909 (5)	3.278 (1)
CFT	fluency	2.806 (4)	2.846 (2)	2.268 (6)	2.255 (7)	2.808 (3)	2.615 (5)	3.023 (1)
Mean rank		2.95	4.33	5.00	4.76	3.43	4.05	3.48

C.3 Evaluator Reliability and Human Validation

ICC denotes the intraclass correlation coefficient. We foreground consistency ICC(C,3), the two-way, average-measures reliability of three ratings after accounting for additive differences in rater severity. We also report absolute-agreement ICC(A,3), which includes these severity differences, to assess agreement on score levels. Both are computed from raw ratings on complete three-scorer units. Across ten first-level metrics, the LLM panel has unweighted macro ICC(C,3) $= 0.792$ and ICC(A,3) $= 0.688$.

Table 20: Three-judge reliability per first-level metric. Consistency ICC(C,3) is the primary measure; absolute-agreement ICC(A,3) is supplementary. The last row reports unweighted macro averages.

Task	First-level item	Complete units	ICC(C,3)	ICC(A,3)
CreativeMath	Correctness	5,902	0.739	0.722
CreativeMath	Coarse novelty	4,005	0.755	0.733
CreativeMath	Fine novelty	916	0.812	0.792
CS4	Constraint count	2,700	0.997	0.996
CS4	Story quality	2,610	0.875	0.826
CS4	Novelty	2,609	0.808	0.749
CFT	Fluency	14,382	0.729	0.563
CFT	Flexibility	14,382	0.743	0.465
CFT	Appropriateness	14,382	0.699	0.516
CFT	Novelty	14,382	0.764	0.516
Unweighted macro over first-level items		–	0.792	0.688

Human Annotation and Human–LLM Agreement. Three annotators each scored 40 CreativeMath outputs, 40 CS4 outputs, and 60 CFT outputs sampled from the Direct outputs of the 11 models. Approximately 70% follow fixed-seed random sampling and 30% are sampled across low, medium, and high LLM-score bands; each source prompt contributes at most one model output. The study materials contain 12 calibration items, which are excluded from the reported statistics. Annotators scored independently under randomized order, with model identities, judge identities, and LLM scores hidden. Human consensus is the three-rater mean, and LLM consensus is the three-judge mean.

Complete counts reflect conditional metric availability: CreativeMath fine novelty has 15 complete human units and ten paired human–LLM units, while CreativeMath correctness is constant at one across the human subset; this subset therefore does not test recognition of incorrect solutions. Across nine estimable metrics, the unweighted macro human reliability is ICC(C,3) $= 0.866$, with supplementary ICC(A,3) $= 0.805$. Per-metric human–LLM Pearson correlations span 0.547–0.998; all four CFT dimensions fall between 0.547 and 0.637. Human–LLM correlation intervals use 2,000 item-bootstrap replicates.

Interpreting Human–LLM Agreement. On the annotated subset, CFT has human ICC(C,3) of 0.742–0.812, compared with 0.831–1.000 for the estimable CreativeMath and CS4 metrics (Table 21). Its human–LLM correlations should therefore be interpreted alongside the lower consistency of the human reference. The rubric requires judgments of representational departure beyond the supplied change, cross-system coherence, and conditional validity under new premises. These demands may contribute to scoring uncertainty, alongside rubric ambiguity, rater calibration, and sample score variation. ICC(A,3) measures the absolute-agreement reliability of the three-annotator mean; ICC(C,3) discounts additive scorer-level offsets. Pearson and Spearman measure association between human and LLM means, while MAE quantifies their absolute score differences.

Table 21: Human reliability and human–LLM agreement on the mixed random and score-stratified 140-output Direct subset. Pearson's r and Spearman's ρ compare the three-annotator mean with the three-judge mean. n_H counts units rated by all three annotators, and $n_{H\text{-}L}$ counts units with both consensus scores. Human ICC(C,3) is the primary reliability measure and ICC(A,3) is supplementary. ICC macro values are unweighted over estimable metrics; MAE (mean absolute error) retains each metric's task-specific scale.

Task	Metric	n_H	$n_{H\text{-}L}$	ICC(C,3)	ICC(A,3)	r	ρ	MAE
CreativeMath	Correctness	40	40	–	–	–	–	0.000
CreativeMath	Coarse novelty	40	40	0.922	0.923	0.803	0.788	0.175
CreativeMath	Fine novelty	15	10	1.000	1.000	0.815	0.823	0.200
CS4	Constraint count	40	40	0.999	0.999	0.998	0.995	0.425
CS4	Story quality	40	40	0.936	0.919	0.920	0.818	0.238
CS4	Novelty	40	40	0.831	0.780	0.831	0.781	0.425
CFT	Fluency	60	60	0.812	0.709	0.637	0.628	0.744
CFT	Flexibility	60	60	0.807	0.613	0.567	0.591	0.436
CFT	Novelty	60	60	0.742	0.688	0.547	0.548	0.317
CFT	Appropriateness	60	60	0.744	0.612	0.611	0.632	0.283
Unweighted ICC macro over estimable metrics		–	–	0.866	0.805	–	–	–

C.4 Full Ablation Matrix

The tested LOO groups combine related atomic operators: *exploration* removes exploration-axis expansion plus coverage balancing; *novelty* removes novelty transformation; *selection* removes portfolio selection, diversity filtering, and semantic deduplication; *rule extraction* removes rule-change extraction plus legacy-assumption mapping; *breakage* removes breakage propagation plus primitive induction; and *reconstruction* removes system reconstruction, performance re-anchoring, and norm/interface establishment. Unit extraction and relation abstraction are ablated individually.

Table 22: Complete LOO operator-group ablation matrix on the matched one-fifth subset at temperature 0.7. Entries are percent change when the group is ablated, $(\text{LOO} - \text{full})/|\text{full}| \times 100$, where LOO and full denote the scores after removing one operator group and with the complete workflow, respectively, and $|\text{full}|$ is the absolute full score; negative values mean that removal lowers the task-specific dimension. The full column is the three-repeat mean before ablation.

Pool	Task	Task-specific dimension	Full	Group 1	Group 2	Group 3
Comb.	RAT	accuracy	0.782	−33.8% unit	−29.4% relation	–
Comb.	BATS	accuracy	0.662	−0.9% unit	−0.9% relation	–
Comb.	Metaphor	fit	0.468	−2.6% unit	+0.2% relation	–
Expl.	AUT	fluency	38.200	−14.8% explore	−2.3% novelty	−29.0% select
Expl.	AUT	flexibility	21.240	−11.5% explore	+12.6% novelty	−25.2% select
Expl.	AUT	originality	47.201	+13.2% explore	+4.2% novelty	+10.7% select
Expl.	AUT	applicability	0.756	+1.6% explore	−3.4% novelty	−3.2% select
Expl.	NeoCoder	fluency	0.586	+42.9% explore	+42.9% novelty	+32.9% select
Expl.	NeoCoder	originality	0.225	−72.0% explore	−77.6% novelty	−81.4% select
Expl.	NeoCoder	appropriateness	0.517	−51.5% explore	−49.1% novelty	−56.3% select
Expl.	CreativeMath	correctness	0.882	+2.2% explore	+2.2% novelty	+2.2% select
Expl.	CreativeMath	coarse novelty	0.568	−34.5% explore	−40.3% novelty	−30.1% select
Expl.	CreativeMath	originality	0.445	−36.5% explore	−41.7% novelty	−30.4% select
Expl.	CreativeMath	appropriateness	0.882	+2.2% explore	+2.2% novelty	+2.2% select
Expl.	CS4	fluency	0.156	+18.7% explore	−54.3% novelty	−58.7% select
Expl.	CS4	flexibility	0.941	−1.2% explore	−1.1% novelty	−1.5% select
Expl.	CS4	novelty	2.260	+21.8% explore	+28.5% novelty	+21.3% select
Expl.	CS4	appropriateness	0.748	+8.8% explore	+11.2% novelty	+8.8% select
Transf.	CFT	fluency	2.823	−0.3% rule	+0.1% breakage	+0.4% rebuild
Transf.	CFT	novelty	3.140	−11.6% rule	−11.2% breakage	−10.6% rebuild
Transf.	CFT	appropriateness	3.286	−3.9% rule	−3.3% breakage	−3.1% rebuild
Transf.	CFT	flexibility	3.171	−1.8% rule	−1.3% breakage	−1.3% rebuild

C.5 Additional Robustness Analyses

C.5.1 CFT Score Distribution and Source Split

Across 14,382 complete Direct model–item units, the deliberately stringent CFT rubric yields a highly selective upper tail. Between 78.5% and 96.1% of scores lie in $[2.5, 3.5]$, and at most 2.34% reach 4.0. Current models therefore seldom satisfy the upper-band requirements for coherent system-wide reconstruction.

Table 23: CFT Direct-score distribution over 14,382 complete model–item units. Central mass is the proportion in $[2.5, 3.5]$; upper-band mass is the proportion at or above 4.0 on the native 0–5 scale.

Dimension	Mean	SD	Q05	Q25	Median	Q75	Q95	Central / upper-band mass
Fluency	2.991	0.327	2.500	2.833	3.000	3.167	3.500	96.1% / 0.01%
Flexibility	3.222	0.312	2.833	3.000	3.167	3.500	3.667	90.0% / 0.67%
Novelty	2.879	0.429	2.167	2.667	3.000	3.167	3.500	83.0% / 0.06%
Appropriateness	3.228	0.420	2.500	3.000	3.333	3.500	3.833	78.5% / 2.34%

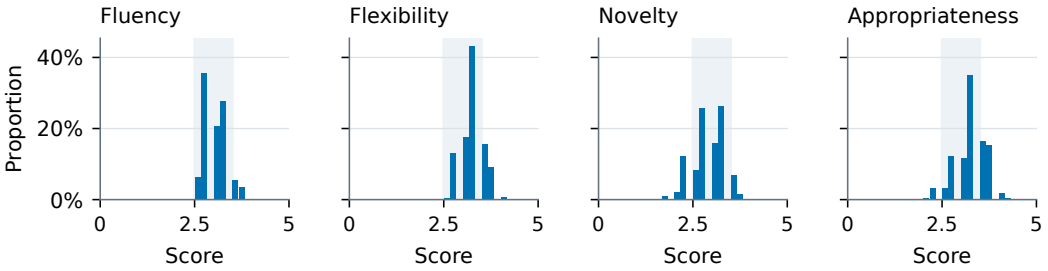


Figure 7: CFT Direct-score distributions across 11 models and 14,382 complete model–item units. Shading marks $[2.5, 3.5]$. The sparse upper tail reflects the rubric's demanding criteria for system-wide reconstruction.

We compare performance across the historical and synthetic scenario sets. Historical-prototype items have lower observed scores than synthetic items. Their novelty and appropriateness means are lower than the synthetic split by 0.364 and 0.229, respectively. Source-specific model rankings remain strongly aligned, reaching Spearman $\rho = 0.882$ for novelty and 0.964 for appropriateness.

Table 24: CFT Direct scores by source. Differences are historical minus synthetic; intervals use 10,000 parent-scenario bootstrap replicates. The last column correlates the 11 source-specific model rankings.

Dimension	Historical	Synthetic	Difference	95% interval	Rank ρ
Fluency	2.895	3.010	−0.115	$[-0.157, -0.072]$	0.982
Flexibility	3.108	3.245	−0.137	$[-0.182, -0.090]$	0.982
Novelty	2.577	2.941	−0.364	$[-0.458, -0.269]$	0.882
Appropriateness	3.038	3.267	−0.229	$[-0.311, -0.148]$	0.964

C.5.2 Judge-Panel Sensitivity

The novelty–appropriateness relation and novelty-based model ordering remain stable after removing any one judge. Across the three leave-one-judge panels, the median within-model correlation ranges from 0.745 to 0.878, and the model ordering has Spearman $\rho \geq 0.982$ relative to the full panel.

Table 25: CFT judge-panel sensitivity over the complete Direct records. N–A r is summarized across the 11 within-model correlations. Rank ρ compares novelty-based model orderings with the full panel.

Panel	Median N–A r	Range	Rank ρ	Maximum rank shift
All three judges	0.861	$[0.811, 0.939]$	1.000	0
Without GPT	0.878	$[0.857, 0.926]$	0.982	1
Without Gemini	0.745	$[0.701, 0.918]$	0.991	1
Without DeepSeek	0.812	$[0.768, 0.929]$	0.991	1

C.5.3 Answer-Length Controls

We compute within-model partial Pearson correlations by separately regressing novelty and appropriateness on an intercept and an answer-length covariate, then correlating the residuals. Length is measured in characters, $\log(1 + L)$ where L is the character count, or whitespace-delimited words. The analysis uses the same complete Direct CFT outputs as the judge-panel analysis; summaries give medians and minimum–maximum ranges across 11 models.

Table 26: Length controls for CFT novelty–appropriateness correlations.

Control	Median within-model r	Across-model range
None	0.861	$[0.811, 0.939]$
Characters	0.854	$[0.808, 0.900]$
Log characters	0.857	$[0.808, 0.875]$
Words	0.853	$[0.809, 0.900]$

The novelty–appropriateness association remains strong after linear length adjustment. Human–LLM agreement is more length-sensitive, as shown by applying the same residualization to the 60 human-rated Direct outputs below. Length can reflect both response completeness and evaluator sensitivity; shared scoring effects are examined in the following analysis.

Table 27: CFT human–LLM correlations before and after length control ($n = 60$).

Dimension	Raw	Characters	Log characters	Words
Fluency	0.637	0.423	0.337	0.441
Flexibility	0.567	0.421	0.372	0.429
Novelty	0.547	0.512	0.490	0.514
Appropriateness	0.611	0.480	0.452	0.484

C.5.4 Human and Cross-Judge Novelty–Appropriateness Associations

On the 60 human-rated CFT outputs spanning 59 parent scenarios, novelty and appropriateness averaged over three annotators correlate at $r = 0.697$. A 3,000-replicate parent-cluster bootstrap (seed 20260920) gives a 95% interval of $[0.549, 0.816]$; controlling log character count yields $r = 0.566$. The subset combines random and score-stratified sampling.

Table 28: Supplementary CFT novelty–appropriateness validation. Human-subset rows pool 60 outputs; full-model rows summarize within-model correlations across 11 models.

Scope	Scoring arrangement	Pearson r
Human subset	Three-annotator mean	0.697
Human subset	Human mean, log-length adjusted	0.566
Human subset	Three-judge mean	0.861
Full model set	Same-judge dimension pairs	0.813
Full model set	Cross-judge dimension pairs	0.355

For the full model set, the same-judge summary is the across-model median of each model's median over three same-judge correlations; the cross-judge summary uses the six ordered pairs of distinct judges in the same way. All cross-judge pair correlations are positive, with per-model cross-judge medians ranging from 0.289 to 0.566. The association extends to human ratings and across judges, while its strength depends on the scoring arrangement. The same- versus cross-judge difference may reflect both shared scoring tendencies and measurement noise, rather than isolating either source.

C.5.5 Routing Structure

Stage 2 produces concentrated, task-specific workflow distributions. Exact agreement with each task's modal ordered workflow averages 73.0% across tasks, and the mean operator-set Jaccard similarity to that modal workflow is 0.956. The observed profiles range from one workflow on RAT to eight on CreativeMath.

Table 29: Task-conditioned routing structure on the full Qwen3.5-27B TCE run. Exact modal is the proportion matching the task-modal ordered workflow; Jaccard ignores order and compares operator sets.

Task	n	Workflows	Exact modal	Mean Jaccard
RAT	144	1	100.0%	1.000
Metaphor	2,953	4	63.6%	0.938
BATS	4,000	2	62.9%	0.938
AUT	73	3	47.9%	0.932
NeoCoder	1,194	2	92.7%	0.985
CreativeMath	605	8	56.7%	0.938
CS4	250	4	60.8%	0.916
CFT	1,308	2	99.2%	1.000
Task macro	–	–	73.0%	0.956

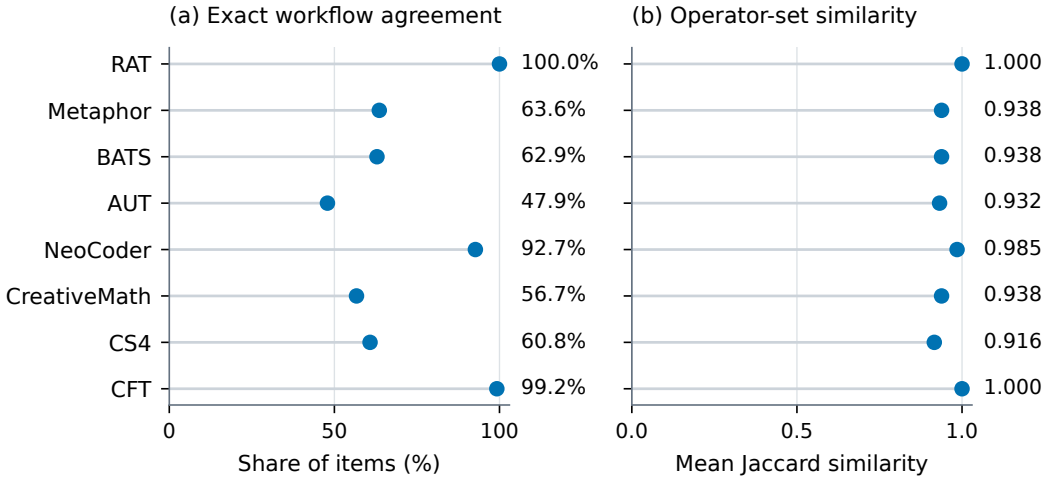


Figure 8: Routing structure. Left: exact agreement with the task-modal ordered workflow. Right: mean Jaccard similarity between each item's operator set and its own task-modal set, ignoring order.

C.5.6 BATS Category Balance

BATS contains 20 equally weighted categories with 200 items each. Gemini leads Claude in 11 categories, trails in 6, and ties in 3, yielding an aggregate margin of 0.0102.

Table 30: BATS category accuracy for the two leading Direct models. Every row contains 200 items. Difference is Gemini minus Claude.

Group	Category	Gemini	Claude	Difference
Encyclopedic	country–capital	0.990	0.985	+0.005
	country–language	0.640	0.660	−0.020
	UK city–county	0.585	0.585	0.000
	name–nationality	0.575	0.580	−0.005
	name–occupation	0.650	0.645	+0.005
	animal–young	0.580	0.590	−0.010
	animal–sound	0.610	0.625	−0.015
	animal–shelter	0.335	0.360	−0.025
	things–color	0.610	0.590	+0.020
	male–female	0.970	0.955	+0.015
Lexicographic	hypernyms–animals	0.590	0.530	+0.060
	hypernyms–misc	0.490	0.465	+0.025
	hyponyms–misc	0.765	0.675	+0.090
	meronyms–substance	0.575	0.555	+0.020
	meronyms–member	0.700	0.715	−0.015
	meronyms–part	0.620	0.600	+0.020
	synonyms–intensity	0.535	0.535	0.000
	synonyms–exact	0.745	0.715	+0.030
	antonyms–gradable	0.870	0.870	0.000
	antonyms–binary	0.840	0.835	+0.005
Overall		0.6637	0.6535	+0.0102

C.5.7 Method-Rank Sampling Stability

We resample independently within tasks, sharing each draw across all methods. NeoCoder uses original problems, CreativeMath uses normalized question-text clusters, CS4 uses stories, and CFT uses parent scenarios (Appendix C.2). AUT originality is recomputed from the sampled top 10%. CS4 first averages constraint levels within each story. In 10,000 replicates (seed 20260916), TCE has the lowest mean rank in 99.86% and is the unique leader in 99.76%. These frequencies describe sampling stability on the evaluated task collection.

Table 31: Seven-method mean ranks and source-cluster bootstrap stability. Lower is better. The interval and Lowest columns refer to the supplementary metric-equal mean rank; Lowest includes ties. Task-equal weighting is supplementary.

Method	Metric-equal rank	Task-equal rank	Bootstrap 95% interval	Lowest	Unique lowest
TCE	2.952	3.208	[2.571, 3.143]	99.86%	99.76%
Self-Refine	3.429	3.958	[3.238, 3.857]	0.22%	0.13%
Direct	3.476	3.760	[3.381, 3.857]	0.02%	0.01%
ToT (1 call)	4.048	4.125	[3.714, 4.286]	0.00%	0.00%
Generic	4.333	5.042	[4.000, 4.524]	0.00%	0.00%
Self-Consistency	4.762	3.719	[4.429, 5.048]	0.00%	0.00%
Structured CoT	5.000	4.188	[4.714, 5.286]	0.00%	0.00%

This rank-based summary weights all 21 task–metric combinations equally. Task-equal weighting first averages metric ranks within each task, then averages the eight task means. TCE remains lowest under this alternative (3.208), followed by Self-Consistency and Direct. Equal-category mean rank is also lowest for TCE (3.060).

C.5.8 Within-Family Dense–MoE Comparison

Figure 9 shows the dense-model trajectories on native metric scales.

The two Qwen3.5 MoE checkpoints complement the three-checkpoint dense scaling comparison. Both Qwen3.5-35B-A3B and Qwen3.5-122B-A10B exceed dense Qwen3.5-9B on all eight selected native fields. Against dense Qwen3.5-27B, the 122B-A10B model leads BATS, RAT, and NeoCoder, while the dense model leads the other five tasks. The architecture comparison therefore produces task-specific profiles inside one model family.

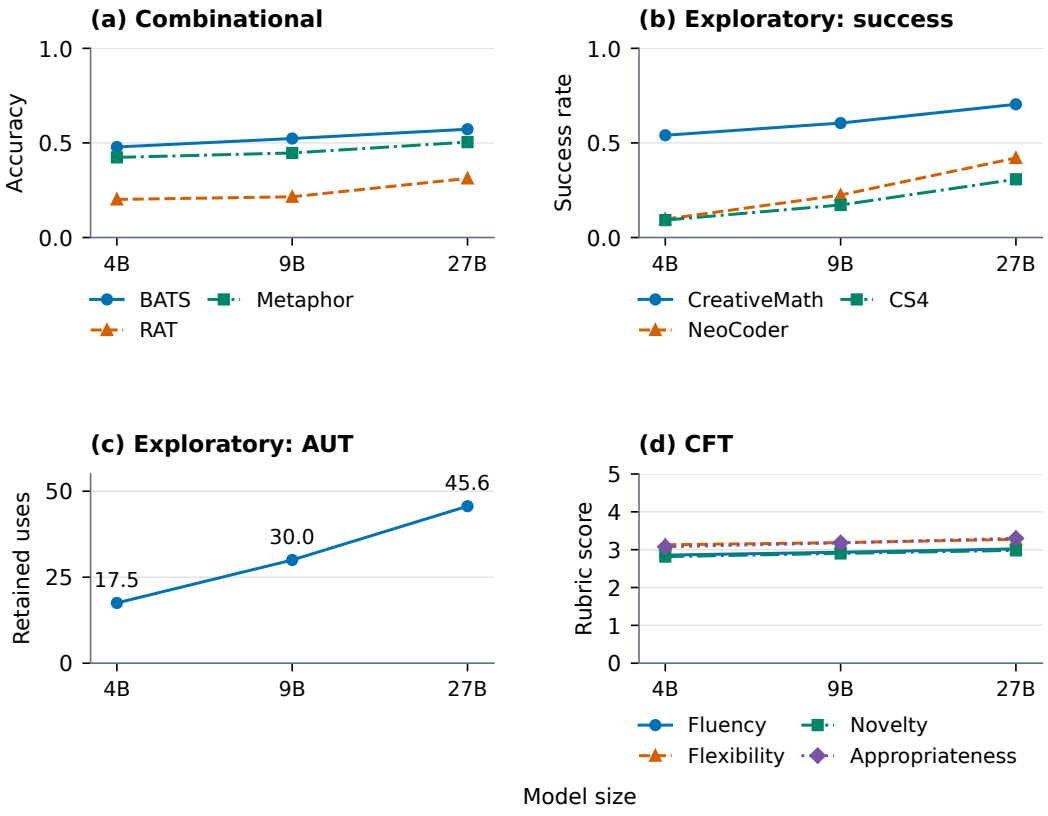


Figure 9: Dense Qwen3.5 scaling under direct prompting. Panels show native accuracy/success rates, AUT counts, and CFT rubric scores. Lines connect observed checkpoints.

For the task composites in Table 2, rather than the selected native fields below, Qwen3.5-122B-A10B exceeds dense Qwen3.5-9B on all eight tasks; 35B-A3B exceeds it on seven, with AUT the exception. Against dense Qwen3.5-27B, 122B-A10B leads BATS and RAT, while the dense model leads the other six tasks. These are descriptive checkpoint comparisons: total parameters, active parameters, and training configuration vary jointly.

Table 32: Dense and MoE Qwen3.5 checkpoints under Direct prompting. Total and active parameter counts are in billions; all task columns retain their primary native scales.

Model	Arch.	Total	Active	BATS	RAT	Metaphor	AUT	NeoCoder	CreativeMath	CS4	CFT
Qwen3.5-4B	Dense	4	4	0.4785	0.2014	0.4236	17.5205	0.0972	0.5405	0.092	2.8558
Qwen3.5-9B	Dense	9	9	0.5232	0.2153	0.4470	30.0000	0.2245	0.6050	0.172	2.9332
Qwen3.5-27B	Dense	27	27	0.5723	0.3125	0.5042	45.6438	0.4204	0.7041	0.308	3.0206
Qwen3.5-35B-A3B	MoE	35	3	0.5457	0.2222	0.4802	36.6712	0.3283	0.6496	0.232	2.9541
Qwen3.5-122B-A10B	MoE	122	10	0.5737	0.4236	0.4903	39.6164	0.4481	0.7008	0.292	3.0022

C.5.9 Additional Call-Budget Controls

Using Qwen3.5-27B throughout, we compare TCE with Best-of-N sampling with model-based selection, repeated Self-Refine, and multi-call tree search. For item i, we determine the logical-call allowance B_i from TCE's workflow records and execution rules. It includes recorded operator and seed calls, per-item routing, the finalizer's call upper bound, and one full shared task-routing call, with a minimum of four calls. Each control item receives the full shared-routing allocation and finalizer upper bound. A logical call denotes one workflow generation or selection stage. Table 33 specifies each method's allocation.

Objective tasks and AUT target their full sets; CreativeMath, CS4, and CFT use fixed approximately one-fifth subsets selected in dataset order. We reuse TCE's generated responses and evaluate all four methods on common items using the same task-specific protocol. CreativeMath, CS4, and CFT use a DeepSeek V3.2 and Gemini 3.1 Pro judge panel; AUT applicability uses its native DeepSeek-only evaluator. Aggregates use items with valid scores for all four methods; Table 34 reports the counts. AUT originality retains top-decile aggregation, and CS4 retains story-first averaging. Normalization uses the method-table divisors and aggregation in Appendix C.1.1.

Table 4 reports the normalized task and overall scores. Tables below provide coverage and native metrics.

Table 33: Call allocation for the budget controls. B_i is the logical-call allowance derived from TCE's workflow records and execution rules for item i.

Method	Allocation rule	Control calls
TCE	Workflow records and execution rules define B_i.	Reference
Best-of-N	Generate $B_i - 1$ candidates, then select once.	B_i
Self-Refine	Start with a draft, then complete critique–revision pairs.	B_i if odd; $B_i + 1$ if even
Multi-call ToT	Allocate calls to planning, expansion, ranking, and final synthesis.	B_i

Table 34: Evaluation sample counts. Planned is the fixed target per method; method columns count valid scored items, and Matched gives the common evaluation items used for all four methods.

Task	Planned	TCE	Best-of-N	Self-Refine	Multi-call ToT	Matched
BATS	4000	4000	4000	4000	4000	4000
RAT	144	144	144	144	144	144
Metaphor	2953	2953	2953	2953	2953	2953
AUT	73	72	73	71	73	70
NeoCoder	1194	1194	1194	1194	1194	1194
CreativeMath	121	121	121	121	121	121
CS4	50	50	50	50	50	50
CFT	262	262	262	262	262	262

Table 35: Native metric scores for the call-budget controls, on matched items. F/Fx/N/A denote fluency/flexibility/novelty/appropriateness; N_c/N_f are coarse/fine mathematical novelty. Bold is best; underline is second best.

Task	Metric	n	TCE	Best-of-N	Self-Refine	Multi-call ToT
BATS	A	4000	**0.588**	0.586	0.503	0.575
RAT	A	144	0.521	0.354	**0.535**	0.278
Metaphor	A	2953	0.482	**0.516**	0.353	0.495
AUT	F	70	**35.029**	13.043	10.414	18.186
AUT	Fx	70	**21.991**	10.775	7.935	12.857
AUT	N	70	**65.878**	17.733	21.059	29.828
AUT	A	70	0.700	0.894	0.853	**0.903**
NeoCoder	F	1194	0.633	**0.781**	0.725	0.667
NeoCoder	N	1194	**0.248**	0.116	0.176	0.210
NeoCoder	A	1194	0.484	0.370	0.518	**0.539**
CreativeMath	A	121	0.950	**0.967**	0.843	**0.967**
CreativeMath	N_c	121	**0.620**	0.380	0.463	0.264
CreativeMath	N_f	121	**0.140**	0.058	0.124	0.041
CS4	F	50	**0.220**	0.180	0.180	0.140
CS4	Fx	50	**0.941**	0.940	0.627	0.940
CS4	N	50	2.270	2.310	**2.370**	2.310
CS4	A	50	0.779	**0.859**	0.840	0.847
CFT	F	262	**3.180**	2.958	3.176	3.149
CFT	Fx	262	**3.323**	3.061	3.249	3.244
CFT	N	262	**3.002**	2.183	2.501	2.495
CFT	A	262	3.834	3.892	3.721	**3.951**

D Dimension Relationships and Output-Level Analysis

D.1 Task Rankings and Within-Task Metric Correlations

Panel (a) correlates task-composite rankings across the 11 Direct models using Spearman ρ. Panel (b) uses native per-item scores to compute Pearson correlations within each model, summarized by the median and minimum–maximum range across the 11 models. Repeated constraint levels share a source problem, story, or parent scenario.

AUT uses cumulative item scores here, including each object's originality sum, not the task-level top-10% summary. Its fluency and flexibility share quantity-dependent terms, so a high correlation partly reflects score construction. CreativeMath assigns zero novelty after a failed correctness gate and evaluates fine novelty only on eligible, coarsely novel answers; its correlations include those structural zeros. CS4 flexibility is the product of 2-, 3-, and 4-gram diversity ratios, not the breadth of plot ideas. F, Fx, N, and A retain the task-native interpretations in Appendix B.2; Nc and Nf are coarse and fine mathematical novelty.

Figure 10b displays all supported within-task metric pairs on a common $[-1, 1]$ scale. Table 36 adds the across-model ranges; these item-level relationships differ from the model-rank correlations in panel (a).

The same-task profile supplies diagnostic information beyond a primary score. AUT separates the amount of production from category coverage and reference-relative originality. NeoCoder's weak correlations separate prohibited-technique compliance, use of a different detected technique, and successful execution. CS4 constraint completion is more related to appropriateness (0.318) than to lexical flexibility (0.020), consistent with these fields observing different aspects of a story. CFT shows positive relations across all four fields, especially fluency–flexibility (0.887) and novelty–appropriateness (0.861). The latter relation is compatible with the need to develop new operational representations when previous procedures become invalid; the historical cases in Appendix E make this interpretation concrete. Shared rubrics and task difficulty can also contribute to the observed associations.

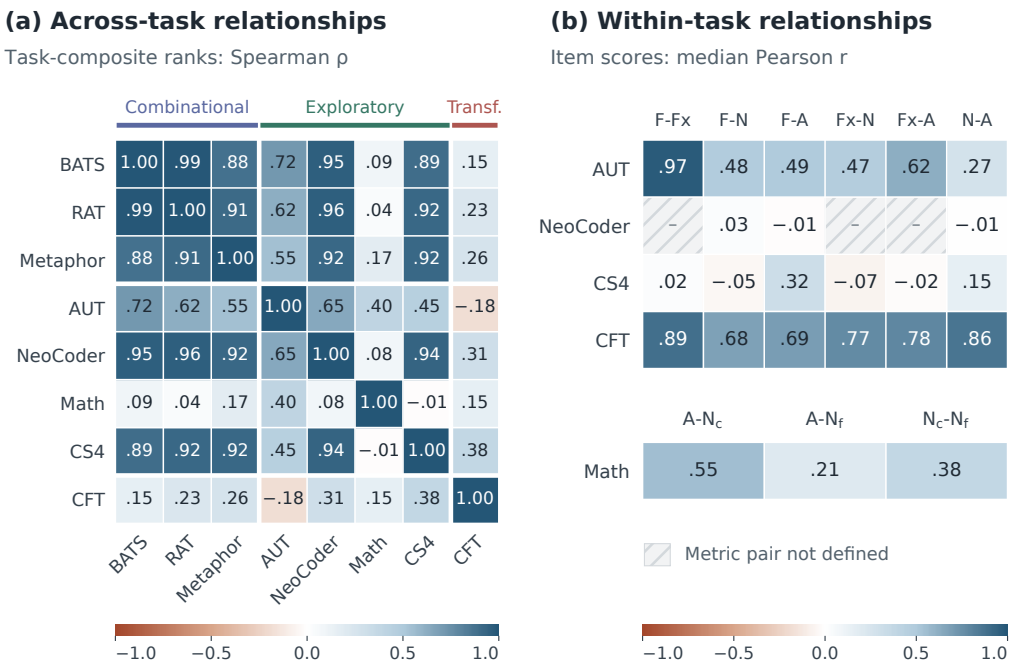


Figure 10: Task and metric relationships under direct prompting. (a) Spearman ρ between task-composite rankings of 11 models. (b) Pearson r between item-level metrics within each model, summarized by the median over the 11 models; all supported pairs are shown. Hatching denotes undefined pairs, not zero correlation. F/Fx/N/A denote fluency/flexibility/novelty/appropriateness; N_c/N_f denote coarse/fine mathematical novelty. Math denotes CreativeMath; CFT denotes Counterfactual Transformation. Metric semantics and scoring dependencies remain task-specific (Appendix D.1).

D.2 Model Cases: What Supports the Task Leaders?

We select paired Direct responses to examine task-specific strengths and failure modes under the same input, constraints, and visible references. Evaluation scores are supplemented by algebraic, code, and constraint checks. In the mathematical cases, k denotes the number of visible reference solutions; all algebraic variables below are local to their respective problems.

M1: Qwen27B versus Claude—A Correct Number Is Not a Complete Proof. AMC 10B 2005, problem 16, $k = 2$, asks about monic quadratics $g(x) = x^2 + px + m$ and $f(x) = x^2 + mx + n$, where x is the polynomial variable, p, m, n are nonzero real coefficients, and the roots of f are twice those of g. The target ratio is n/p. In the comparison below, y is the variable used after rescaling, and r, s denote the roots of g.

	Qwen3.5-27B	Claude Opus 4.6
Core step	$4g(y/2) = y^2 + 2py + 4m = f(y)$, hence $m = 2p$, $n = 4m$, and $n/p = 8$.	Subtracts equations evaluated at roots r, s, then divides by $r - s$. Notes that the equal-root case would require separate verification but does not supply it.
Scores (A,Nc,Nf)	$(1, 1, 0)$	$(0, 0, 0)$
Boundary check	The polynomial transformation preserves multiplicity.	$p = 8, m = 16, n = 64$ gives $g = (x+4)^2$ and $f = (x + 8)^2$: a valid equal-root case omitted by the derivation.

Both responses end with 8. Qwen's structural transformation avoids the distinct-root assumption; this explains why correctness of the proof separates outputs that share a final answer. Its coarse

Table 36: Metric-pair correlations: medians and minimum–maximum ranges across the 11 models.

Task	Pair	Median r	Model range
AUT	F–Fx	0.970	[0.941, 0.995]
AUT	F–N	0.482	[0.185, 0.676]
AUT	F–A	0.495	[0.294, 0.830]
AUT	Fx–N	0.471	[0.185, 0.656]
AUT	Fx–A	0.624	[0.362, 0.882]
AUT	N–A	0.274	[0.119, 0.581]
NeoCoder	F–N	0.030	[-0.138, 0.155]
NeoCoder	F–A	-0.015	[-0.160, 0.108]
NeoCoder	N–A	-0.007	[-0.124, 0.148]
CreativeMath	A–Nc	0.551	[0.449, 0.905]
CreativeMath	A–Nf	0.208	[0.178, 0.321]
CreativeMath	Nc–Nf	0.378	[0.345, 0.426]
CS4	F–Fx	0.020	[-0.176, 0.209]
CS4	F–N	-0.046	[-0.142, 0.032]
CS4	F–A	0.318	[0.165, 0.456]
CS4	Fx–N	-0.065	[-0.204, 0.055]
CS4	Fx–A	-0.018	[-0.147, 0.144]
CS4	N–A	0.154	[0.052, 0.673]
CFT	F–Fx	0.887	[0.823, 0.971]
CFT	F–N	0.678	[0.629, 0.884]
CFT	F–A	0.692	[0.653, 0.925]
CFT	Fx–N	0.771	[0.728, 0.899]
CFT	Fx–A	0.783	[0.744, 0.943]
CFT	N–A	0.861	[0.811, 0.939]

novelty is relative to the shown root-wise reference solutions, while fine novelty remains zero. Thus correctness and method difference convey distinct information.

M2: Simultaneous Constraints, Not a Plausibility Argument. For AMC 10A 2004, problem 14, $k = 2$, let V be the total value in cents and N the number of coins. The two average-value conditions give $V = 20N$ and $V + 25 = 21(N + 1)$, hence $N = 4$ and $V = 80$. Qwen uses the resulting small integer system, including the pennies' congruence modulo five, to obtain three quarters and one nickel. Claude's alternative derivation introduces spurious solutions and appeals to a small purse without eliminating them with both average conditions. Substituting the proposed 34-quarter, 5-nickel, 5-penny alternative gives $(880+25)/(44+1) \neq 21$. This exposes a checkable loss of constraint equivalence.

M3: The Reverse Contrast—Claude Closes the Integer Cases. In IMO 1977, problem 5, $k = 1$, the unknown natural numbers a, b satisfy $a^2 + b^2 = q(a + b) + r$ and $q^2 + r = 1977$, where q and r are the quotient and remainder, with $0 \leq r < a + b$. Claude derives $q = 44$, $r = 41$ and completes $(a - 22)^2 + (b - 22)^2 = 1009 = 15^2 + 28^2$, recovering $(7, 50), (50, 7), (37, 50), (50, 37)$. Qwen initially simplifies the parity of $s^2 - sq + q^2$ incorrectly, where $s = a + b$, but later corrects that error and also obtains $q = 44$. Its final integer classification nevertheless omits $(7, 50)$ and $(50, 7)$. Correctness scores favor Claude. Across the 605 matched outputs, the complete joint counts are 351 both correct, 75 Qwen-only correct, 43 Claude-only correct, and 136 both incorrect. Together, these contrasts identify structural reasoning and exhaustive boundary checks as concrete behaviors relevant to mathematical correctness.

M3b: Valid Solutions Can Differ in Computational Organization. For AMC 10A 2014/3, $k = 1$, both models correctly obtain a bakery profit of 52. Qwen sums batch margins, $24(2.50 - 0.75) + 16(1.25 - 0.75) + 8(1.00 - 0.75) = 52$. Claude computes the weighted average selling price $88/48 = 11/6$ and then $48(11/6 - 3/4) = 52$. Their (A, N_c, N_f) scores are $(1, 0, 0)$ and $(1, 1, 0)$, respectively. The outputs differ in how the same valid arithmetic is organized: batch contributions versus an average-price representation. The coarse judgment is relative to the shown references, and neither receives fine novelty. This complements the proof-failure cases by showing what method-sensitive evaluation adds when both answers are valid.

M4: Code—A Valid Alternative Can Be Shared by Multiple Strong Models. For NeoCoder 1894A at constraint depth 5, both Claude and Gemini replace a conventional loop form with a `while` loop and exploit the fact that the final character determines the winner:

```
# Claude: retained solution excerpt
t = int(input())
i = 0
while i < t:
    n = int(input())
    s = input()
    print(s[-1])
    i += 1
```

Both have compliance 1, one newly detected technique, and correctness 1. The case illustrates how preserving a problem invariant supports a concise, correct alternative implementation under technique constraints.

M5: AUT—Quantity and Coverage Do Not Imply Reference-Relative Originality. On *brick*, the first round retains 38 Claude uses versus 13 Qwen uses; their flexibility scores are 17.844 and 10.817, while both originality scores are zero. Claude's response extends across support/weight, garden structures, thermal uses, surface operations, and decorative objects; Qwen's shorter response includes doorstop, bookend, tarp weight, light anvil, mosaic canvas, and leveling support. These content descriptions illustrate differences in production; flexibility scores use the evaluator's reference categories. Claude broadens production and coverage here without exceeding the originality threshold. This contrasts use count with what is new relative to the task references; the primary model table uses cumulative rounds rather than this first-round illustration.

M6: CS4—Preserve the Causal Relations between Required Events. On story 2 with 39 constraints, the protagonist must encounter danger and lose skiing equipment. Qwen writes that the protagonist *left my skis behind to hike*; GPT-5.2 instead describes bindings snapping and skis sliding into a crevasse after an accident. The latter integrates the loss into the required causal sequence. Claude also narrates an accident but does not achieve full completion of all 39 requirements. The $(\mathrm{F}, \mathrm{Fx}, \mathrm{N}, \mathrm{A})$ scores are $(0, 0.923, 2.667, 0.444)$ for Qwen, $(0, 0.961, 3.000, 0.867)$ for Claude, and $(1, 0.934, 3.667, 0.933)$ for GPT-5.2. Claude's higher lexical diversity does not imply full constraint completion; GPT's stronger performance on this item combines explicit event coverage with relevance and coherence.

M7: Reconstruction—Link Representations to Downstream Operation. The paired non-Euclidean, relativistic, and quantum examples in Appendix E complement the cases above. Their informative differences concern whether a changed rule reaches the representation, interfaces, validation, and operating conventions of the rebuilt system. This connects the four-dimensional reconstruction profile to concrete representations and their operational consequences.

D.3 Generic Baselines: Metric Changes and Process Evidence

All method cases hold Qwen3.5-27B and the visible item fixed. Figure 11 shows generic-baseline differences from Direct using the native scores in Table 19, with first-round AUT fields. Tables 37 and 38 analyze paired execution traces.

B1: RAT—Candidate Agreement Can Recover a Shared Relation. For item 64, *dress/dial/flower*, Direct's *rose* fails the exact-match target. Self-Consistency's three candidates all select *sun* and explicitly recover sundress, sundial, and sunflower; accuracy changes $0 \rightarrow 1$. Over the 144 paired RAT items, 61 change from incorrect to correct, one changes from correct to incorrect, 44 stay correct, and 38 stay incorrect. The operation is candidate generation plus agreement-based selection; the observed improvement does not isolate the effect of voting from that of the changed generation prompt.

For canonical answers, exact normalization can aggregate shared solutions. On all retained AUT, NeoCoder, CS4, and CFT pairs, however, three distinct outputs trigger a tie and select the first candidate. This is an implementation-specific selection limitation: voting over exact strings cannot

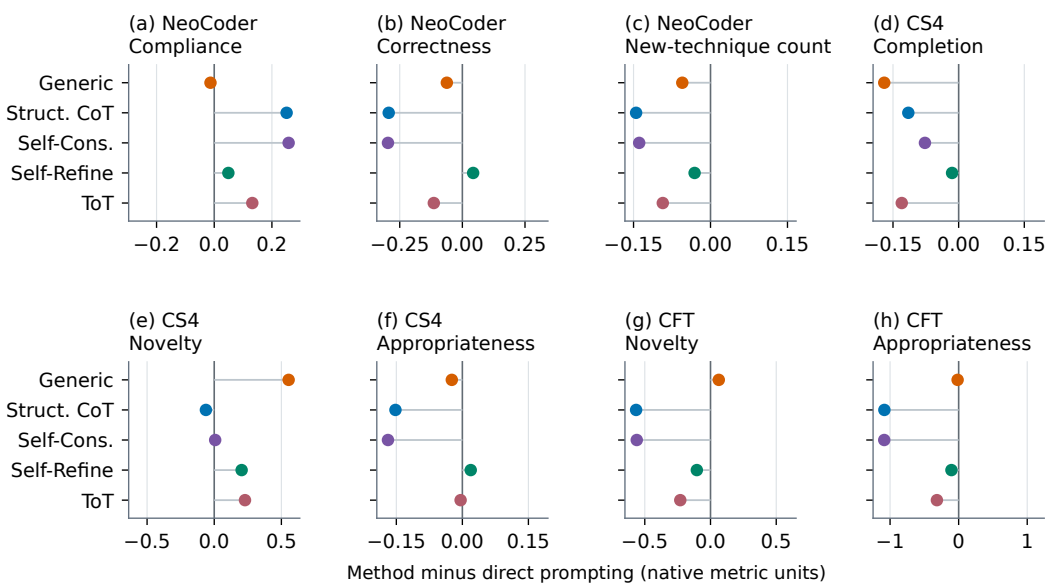


Figure 11: Selected generic-baseline differences from Direct, using the native scores in Table 19. Zero denotes no change; each panel has its own metric scale and all panels share the same method order.

Table 37: Selection behavior in retained three-candidate Self-Consistency traces. Unique means all three normalized candidates differ; first means the selected candidate is index zero.

Task	Paired traces	Majority	Unique	First
BATS	4000	3621	379	3647
RAT	144	104	40	133
Metaphor	2953	2636	317	2618
AUT	73	0	73	73
NeoCoder	1194	0	1194	1194
CreativeMath	604	207	397	559
CS4	197	0	197	197
CFT	1035	0	1035	1035

compare the quality of semantically different open constructions. It motivates checking candidate feasibility and using a task-appropriate equivalence or quality criterion.

B2: AUT—List Expansion and Originality Are Distinct Gains. On *brick*, Self-Refine's critique requests less conventional uses, and the revision inserts suggestions such as an acoustic diffuser, fermentation weight, and texture stamp. The raw list grows from 49 draft entries to 91 revised entries, of which the evaluator retains 37 uses. Relative to Direct's first round, retained-use count changes 13 $\rightarrow$ 37, flexibility 10.817 $\rightarrow$ 28.618, and originality 0 $\rightarrow$ 0.057. Many revised entries reuse *base for a DIY ...* and *weight for a DIY ...* constructions. The trace therefore distinguishes list expansion, retained uses, and reference-relative originality. Reported metric changes compare the final revision with the matched Direct round.

B3: NeoCoder—Branches Repeat an Omitted Boundary Condition. Problem 1851A at depth 2 prohibits `if` statements and dictionaries. Direct and ToT both have compliance 1 and new-technique count 0, but correctness changes $1 \rightarrow 0$.

```
# Direct: h is the list of other people's heights.
sum((abs(H-x) % k == 0)
    and (1 <= abs(H-x)//k <= m-1) for x in h)

# Single-call ToT: evaluated code, inside the loop.
diff = abs(x-H)
if_diff_mod = (diff % k == 0)
if_diff_div = (diff//k < m)
count += if_diff_mod * if_diff_div
```

The selected program counts `diff=0`, although the two people must stand on different steps. Direct's recorded outputs are $[2, 1, 4, 1, 0, 0, 3]$; ToT produces $[2, 1, 5, 1, 1, 0, 3]$. Its other branches reuse the same incomplete predicate, and the comparison favors syntax compliance without finding the missing lower bound. More branch descriptions therefore do not automatically provide more independent validity checks. Across all 1,194 paired items, ToT repairs 88 Direct failures but breaks 225 correct outputs (277 remain correct and 604 remain incorrect).

B3b: Structured CoT—Repair Compliance While Preserving the Solution. On NeoCoder 1702A at depth 1, the banned technique is an `if` statement. Direct computes the correct price reduction $m - 10^{\lfloor \log_{10} m \rfloor}$, where m is the positive integer price, $\log_{10}$ is the base-10 logarithm, and $\lfloor \cdot \rfloor$ denotes the floor function, but includes an `if not input_data` guard. Structured CoT uses the same digit-length invariant with line-wise input and removes that guard. Compliance changes $0 \rightarrow 1$, while correctness remains 1 and new-technique count remains 0. The case demonstrates a targeted compliance repair that preserves the valid solution.

B4: CreativeMath—Selection for a Different Route Must Preserve Execution. For AMC 12A 2007/18, $k = 1$, the monic quartic $f(x) = x^4 + ax^3 + bx^2 + cx + d$ has real coefficients a, b, c, d and roots $2i$ and $2 + i$, where x is the polynomial variable and i is the imaginary unit ($i^2 = -1$). The target is $a + b + c + d$. Direct constructs $f(x) = (x^2 + 4)(x^2 - 4x + 5)$ and obtains 9. ToT proposes a potentially useful shortcut, evaluating the factored polynomial at 1, and selects it for novelty; but it neither carries out $f(1) = 5 \cdot 2 = 10$ nor subtracts the leading coefficient. Its final answer is 1, and correctness changes $1 \rightarrow 0$. Both coarse and fine novelty are zero after the gate. The failure lies in carrying the selected representation through the required computation.

Table 38: CreativeMath correctness transitions in method–Direct comparisons; n is the number of paired items.

Method	n	Wrong→right	Right→wrong	Both right	Both wrong
Generic	577	29	73	329	146
StructCoT	577	28	83	319	147
SelfConsistency	577	25	84	318	150
SelfRefine	577	29	83	319	146
ToT	577	21	101	301	154

A complementary Self-Refine case (AMC 12B 2014/2, $k = 1$) preserves the correct final number 36 but changes the draft's *one-third off* description to *two-thirds off* while still computing a price of 2 from 3. Two judges accept the result and one rejects the inconsistent discount statement, giving a final correctness score of zero under the all-judge gate. The revision is complete: the loss concerns statement consistency, not a wrong final number or truncation. The draft prompt also combines *Keep internal reasoning private* with *Show your reasoning step by step*; its critique treats the explanation as a format violation. This instruction conflict is part of the evaluated implementation and affects interpretation of the revision trace. Aggregate fine-novelty gains should likewise be read alongside the number of answers passing correctness and coarse-novelty gates.

B5: CS4—A New Narrative Angle Can Alter a Required Relation. For story 19 with 39 constraints, Generic Creativity reframes the scene as a surreal trial. Novelty increases $3.000 \rightarrow 3.333$ and lexical flexibility $0.925 \rightarrow 0.950$, while full completion changes $1 \rightarrow 0$ and appropriateness $0.800 \rightarrow 0.693$. Direct has the protagonist spy on three birthday-hat-wearing mice; the revised story says the three mice are *spying on me*, reversing the required roles (constraint 16). It also assigns the high-pitched squeal to the mice rather than the creature (constraint 8). All three judges reject constraint 8 and two reject constraint 16; their satisfied-constraint counts change from $39, 39, 39$ to $38, 35, 35$. This complete response shows how a new narrative angle can lose semantic relations despite retaining the relevant entities. Its lexical diversity rises, so this case explains constraint loss rather than the aggregate flexibility decline. The evaluated response includes the introductory creative-angle explanation as well as the story; the reported scores apply to that full response.

B6: CFT—A Counterexample to an Average Decline. On the royal zero-ledger item with all five active rules (`AX-FM-001-SAMPLE-01-C5-R1R2R3R4R5`), single-call ToT changes $(\mathrm{F}, \mathrm{Fx}, \mathrm{N}, \mathrm{A})$ from $(2.500, 2.833, 2.167, 2.500)$ to $(2.500, 2.667, 2.333, 2.667)$. Both responses distinguish computed zero from unresolved records. Direct nevertheless allows a fallback to positive-only recording and migrates crossed-out debts to zero from historical context. ToT instead keeps a blank unresolved, requires a supervisor annotation and data-retrieval request, and demands a transaction history supporting each zero. These changes preserve the new rule more consistently and accompany a modest appropriateness increase. Its shorter portfolio concentrates on state notation, summaries, training, and audit, while omitting some of Direct's cross-office links; flexibility falls. The case illustrates a trade-off between stronger rule preservation and narrower cross-system coverage.

B7: CFT—Response Allocation Under a Per-Call Limit. Under a fixed per-call output limit, intermediate reasoning can leave less room for completing and validating the final construction. Unlike the complete cases above, the following CFT response leaves its proposal unfinished.

On the five-rule freight-gateway item (`AX-ES-001-SAMPLE-01-C5-R1R2R3R4R5`), CoT's extracted answer ends mid-sentence at `Legacy terms "load`, leaving terminology requirements and subsequent performance validation unfinished. Self-Refine's final answer develops terminology enforcement, performance targets, migration checks, and monitoring triggers. Appropriateness scores are 1.833 and 3.167, respectively. The revision completes terminology enforcement and validation steps left unfinished in the CoT response.

Contributions of Creativity-Specific Operator Groups. The matched one-fifth ablation subset (Table 22) separates three useful effects. Removing unit or relation operators reduces RAT accuracy by 33.8% or 29.4%, consistent with their role in identifying components and shared relations. Removing exploratory groups in NeoCoder raises compliance by 32.9%–42.9% while lowering new-technique count by 72.0%–81.4% and correctness by 49.1%–56.3%: easy syntactic compliance can coexist with weaker solution construction. Removing CFT rule, breakage, or rebuild groups reduces novelty by 10.6%–11.6% and appropriateness by 3.1%–3.9%, with little fluency change. These contrasts support task-specific checks on relation construction, feasible alternatives, and rule propagation.

Implications for Method Improvement. The evidence locates five testable directions: select open candidates using feasibility and semantic quality rather than exact-string agreement; diversify the assumptions checked by branches; preserve verified constraints during revision; carry a selected representation through computation or operational validation; and allocate output space to complete the final proposal.

D.4 Non-Historical Cases: Supplied Premises and Model-Added Representations

These Qwen3.5-27B TCE examples cover physics and economic institutions. Both come from synthetic scenarios without historical-prototype descriptors. We distinguish supplied premises from model-added designs and report three-judge mean fluency, flexibility, novelty, and appropriateness scores (F, Fx, N, A).

S1: Physics—Temperature-Grade Heat Service (Figure 2). Item `AX-SYN-NP-003-ITEM-08-C1-R1` stipulates that mechanical work cannot compensate for an insufficient source temperature: usable heat delivery requires the source to exceed the target by a fixed positive margin. Heat-pump upgrading of waste heat fails, requiring a replacement hospital heating network and temperature-grade operating standards.

Beyond the requested grade-specific standards, the response introduces a *type-safe thermal compiler*: temperature grades become immutable types, implemented through three physically separated circuits. Its *grade-compliant energy accounting* counts below-grade delivery as zero service. The representation links circuit topology and interlocks to metering, dispatch, contracts, and backup supply.

Its proposed interlock cuts power only below the target temperature *minus* $15\,^{\circ}\mathrm{C}$, leaving the required positive margin unenforced. The organizing representation and its local validity checks therefore need separate inspection. Its (F, Fx, N, A) scores are $(2.833, 3.333, 3.333, 3.333)$.

S2: Economics—Path-Dependent Money. Item `AX-SYN-SI-002-ITEM-08-C2-R1R2` makes the discharge value of equal face-value wallet units depend on their institutional-validator sequence. A second rule requires machine-verifiable chain-class certificates and discounted settlement for uncertified segments. The setting is a shared transit-and-retail wallet.

The response develops the certificate requirement into signed *value segments* carrying face amount, validator hash chain, class identifier, and expiry. It separates rapid local authorization from next-day valuation and clearing, then links chain integrity to user-facing value indicators, validator liability, and audit records. These additions coordinate a path-dependent representation of monetary value with exchange procedures and institutional oversight.

Its audit trigger compares total face amount with total realizable value without separating rule-consistent discounts from settlement errors: verification must also adopt the new value semantics. Its (F, Fx, N, A) scores are $(3.167, 3.500, 3.333, 3.667)$. Across the two domains, the cases connect model-added representations to both downstream operations and concrete consistency checks.

E Historically Grounded Case Studies

The adapter passes the complete item JSON, including historical-prototype metadata, into the generation prompt. These cases examine how systematically a response propagates a supplied, historically labeled rule change through representations, operations, validation, interfaces, and norms. Scores are three-judge means. We select them as qualitative illustrations of a clear within-item score contrast and two additional paradigm-level domains. Square brackets in response excerpts indicate editorial clarifications.

E.1 High and Low Reconstruction of Non-Euclidean Geometry

Candidate `AX-FM-002-SAMPLE-01-C2-R1R2` is modeled on non-Euclidean axioms and their later use in general relativity. R1 makes Euclidean geometry a local approximation and requires geodesic behavior under region-specific curvature. R2 requires reference-zone metadata and a curvature model at every merge, adjustment, and route-optimization step.

Higher-Scoring GPT-5.2 Reconstruction (Response Excerpt).

> Replace the "one plane grid" assumption with a **geodetic reference stack**: [a] Global reference frame (GRF) [for] storage and cross-zone computation; [a] curvature model library [with] distortion bounds and validity polygon; [and] zones [where] a particular curvature model and projection is valid within tolerance.
>
> Authoritative coordinates are stored in GRF (3D), plus height type metadata and uncertainty. Local "grid" coordinates become views with declared zone metadata. Parcel legal descriptions store [each] boundary as a sequence of GeodesicSegments (zone-tagged), authoritative corner coordinates in GRF with uncertainties, and a presentation projection for human-readable maps.

Lower-Scoring Qwen3.5-4B Reconstruction (Response Excerpt).

> The fundamental shift is the replacement of the **Single Plane Grid** with a **Hierarchical Geodesic Mesh**. The system operates on a dynamic overlay of local coordinate systems defined by WGS84 ellipsoidal geometry.
>
> All spatial computations now execute on a geodesic solver (e.g., Vincenty, Karney algorithm) [and] before any data leaves a zone's registry or enters a central database, it passes through a Zonal Converter.

Model	F	Fx	N	A	Mean
GPT-5.2	3.5000	3.8333	3.3333	3.6667	3.5833
Qwen3.5-4B	2.5000	2.8333	2.1667	2.3333	2.4583

The higher-scoring answer changes the authoritative spatial primitive, attaches model validity and uncertainty to every representation, and propagates those choices into parcel records, compilation contracts, route optimization, and legal evidence. The lower-scoring answer recognizes curvature and adds geodesic tools, zones, and converters, while concentrating the reconstruction in a familiar WGS84/GIS implementation. In this selected pair, the score gap accompanies a system-wide reconstruction versus a localized technical translation.

E.2 Additional High-Scoring Historical Anchors

E.2.1 General Relativity

Candidate `AX-NP-004-SAMPLE-01-C2-R1R2` is modeled on the shift from Newtonian gravitation to general relativity. Cross-elevation time and distance comparisons become frame- and potential-dependent, and every cross-site measurement records station potential, reference frame, and compensation.

> Under R1, every cross-station comparison must pass through a relativistic transformation. Under R2, every measurement record must carry its station potential, reference frame, and compensation model before it can be used across sites.
>
> Every record in every agency database must carry two layers: [a] `local_value` [in the] station proper frame [and a] `network_value` transformed to [the] regional coordinate frame, with [a] `transform_audit`.

Claude Opus 4.6 propagates frame dependence into a geopotential reference, proper-to-coordinate conversion, dual-layer records, legacy-device edge translation, and cross-agency validation. Its scores are F $= 3.6667$, Fx $= 3.8333$, N $= 3.6667$, and A $= 4.0000$.

E.2.2 Quantum Hypothesis

Candidate `AX-NP-007-SAMPLE-01-C2-R1R2` is modeled on Planck's 1900 quantum hypothesis. Its rules replace arbitrarily divisible radiative exchange with integer energy quanta and condition calibration claims on certified temperature–wavelength ranges.

> Replace "smooth-continuum emission curve + interpolation" with a **quantum-consistent spectral radiance law** and an explicit measurement equation [grounded in] discrete energy quanta $E = nh\nu$.

> [Use] a **forward measurement model** for each instrument class, then solve for temperature (or radiance) with traceable uncertainty. [Publish] modular reference artifacts [whose] certified domains overlap.

In the quoted equation, E is the exchanged energy, n the integer number of quanta, h Planck's constant, and ν the radiation frequency.

GPT-5.2 propagates quantized exchange into the reference law, source corrections, instrument response, inversion, certificate validity domains, uncertainty budgets, and inter-laboratory comparison. Its scores are F $= 3.5000$, Fx $= 3.6667$, N $= 3.0000$, and A $= 3.6667$.

Across these higher-scoring examples, the response replaces an authoritative primitive, derives new validity conditions, and carries them through records, interfaces, validation, and institutions. The lower-scoring response captures the changed vocabulary and core tool choice with a narrower chain of downstream consequences. In these selected examples, the score bands illustrate different depths and breadths of historical-rule reconstruction.

F Workflow Implementation Details

F.1 Generation Settings

Table 39 summarizes the Qwen3.5-27B elicitation configuration. These runs disable the model's separate thinking mode (`enable_thinking=false`); CoT and other prompted reasoning still appear in the generated response.

Table 39: Qwen3.5-27B elicitation settings. Output limits are per-call token caps.

Method / call stage	Temperature	Output cap
Generic Creativity, Structured CoT, single-call ToT	0.7	2,048
Self-Consistency: each sampled response	0.9	2,048
Self-Refine: draft and revision	0.7	2,048
Self-Refine: critique	0.3	2,048
TCE: task-level routing	0.0	256
TCE: item-level operator selection	0.2	1,400
TCE: generative operators	0.7	2,048
TCE: verification	0.0	2,048
TCE: final synthesis (AUT / other tasks)	0.7 / 0.0	2,048

TCE's initial candidate calls use temperatures $(0.0, 0.2, 0.4)$ for BATS and NeoCoder, $(0.7, 0.9, 1.0)$ for CS4, and 0.0 for the remaining tasks. Their output cap is 4,096 for AUT and NeoCoder and 2,048 otherwise. The run-level generation temperature overrides operator-profile defaults; verification and initial-candidate temperatures retain the settings above.

F.2 Information Available to the Workflow

For task T and item x, the visible item view $v(x)$ contains the task prompt, stated constraints, task-visible reference material where supplied, and output schema. The CFT prompt serializes the benchmark record, including origin metadata and a historical-prototype descriptor for 220 items. Task-level metadata q_T contains the task name, output type, and method-defined objective labels and priorities; the method configuration also supplies the operator catalog and inference budgets. Stage 1 converts q_T into creativity-demand profile P_T and routed pool z_T. Stage 2 uses P_T and $v(x)$ to select workflow $W_{T,x}$. Hidden answers, held-out references, target words, evaluator-only candidate lists, judge rubrics, scores, and judge feedback remain evaluation artifacts. The same item-field filter is applied before each workflow call.

With fixed base model B, the Direct and TCE procedures, π_{Direct} and π_{TCE}, produce the following responses for item x of task T:

$$\pi_{\text{Direct}}(B, T, x) = B(v(x)), \tag{3}$$

$$P_T = p(q_T), \qquad z_T = r(P_T), \tag{4}$$

$$W_{T,x} = s(z_T, P_T, v(x)), \tag{5}$$

$$\widetilde{y}_{T,x} = \text{execute}(W_{T,x}, B, v(x)), \tag{6}$$

$$\pi_{\text{TCE}}(B, T, x) = \text{normalize}(\widetilde{y}_{T,x}), \tag{7}$$

$$\Delta_{T,j} = M_{T,j}(B, \pi_{\text{TCE}}) - M_{T,j}(B, \pi_{\text{Direct}}). \tag{8}$$

Here, $B(v(x))$ denotes one Direct model call on the visible input. The functions p, r, and s perform task profiling, pool routing, and ordered workflow selection, respectively; q_T, P_T, z_T, and $W_{T,x}$ are the metadata, demand profile, pool identifier, and workflow defined above. The intermediate artifact $\widetilde{y}_{T,x}$ is returned by execute before task-specific finalization by normalize, which selects or synthesizes the complete answer and enforces the output format. Finally, j indexes a native report field, $M_{T,j}(B, \pi)$ is its task-aggregated score under base model B and response procedure π, and $\Delta_{T,j}$ is the TCE-minus-Direct difference on matched evaluation items.

F.3 Operator Pools and Workflow Selection

Operator Pools. Stage 1 profiles and routes each task at temperature 0; Stage 2 selects item-level operators at 0.2. Item-level selection combines candidate expansion, coverage, and validity checks (Appendix G.2).

The combinational, exploratory, and transformational pools contain seven, ten, and ten routable operators, respectively. Table 40 provides the complete pool membership and role index. The shared `output_normalization` finalization step selects or synthesizes a task-ready answer and enforces the format contract. It is always appended and is excluded from routing and ablation. Appendix H.1 specifies the execution template and the locations of the complete instructions.

G Prompts and Scoring Rubrics

This appendix provides baseline, routing, and three-judge scoring templates. Appendix H provides the shared execution template, complete operator index, and representative instructions; full operator instructions and task-specific overlays are in the anonymous code supplement at the paths listed in Appendix H.1. Prompt wording and placeholders are preserved; line wrapping is adapted for display.

G.1 Three-Judge Scoring Prompts

The three judges are `deepseek-v3.2`, `gpt-5.4`, and `gemini-3.1-pro-preview`. CreativeMath and CS4 use a single user message per scoring call; CFT uses a system message and a user message. The templates follow.

G.1.1 CreativeMath Judge Templates

Correctness User Template (Verbatim).

```
Given the following mathematical problem:
{problem}

Reference solutions:
{reference_solutions}

New solution:
{new_solution}

Please respond ONLY with a JSON object in this exact format:
{"verdict":"YES","error":""}
or
```

```
{"verdict":"NO","error":"<brief reason for the error>"}

Output YES only if the new solution is mathematically correct and
leads to the same result as the reference solutions. Do not include
any extra text.
```

Coarse-Novelty User Template (Verbatim).

```
Criteria for evaluating the novelty of a new mathematical solution
include:
1. If the new solution used to arrive at the solutions is
fundamentally different from reference solutions, such as algebraic
manipulation versus geometric reasoning, it can be considered novel;
2. Even if the final results are the same, if the intermediate steps
or processes involved in reaching those solutions vary significantly,
the new solution can be considered novel;
3. If the new solution relies on different assumptions or conditions,
it should be considered novel;
4. A solution might generalize to a broader class of problems, while
another solution might be specific to certain conditions. In such
cases, they are considered distinct;
5. If the new solution is significantly simpler or more complex than
the others, it can be regarded as essentially novel, even if they lead
to the same result.

Given the following mathematical problem:
{problem}

Reference solutions:
{reference_solutions}

New solution:
{new_solution}

Please respond ONLY with a JSON object in this exact format:
{"verdict":"YES","error":""}
or
{"verdict":"NO","error":"<brief reason for the lack of novelty>"}

Output YES only if the new solution is novel compared to the reference
solutions. Do not include any extra text.
```

Fine-Novelty User Template (Verbatim). The fine-novelty template renders `reference_solutions` from solutions $k+1{:}n_{\text{ref}}$; the coarse template uses $1{:}k$.

```
Criteria for evaluating the novelty of a new mathematical solution
include:
1. If the new solution used to arrive at the solutions is
fundamentally different from reference solutions, such as algebraic
manipulation versus geometric reasoning, it can be considered novel;
2. Even if the final results are the same, if the intermediate steps
or processes involved in reaching those solutions vary significantly,
the new solution can be considered novel;
3. If the new solution relies on different assumptions or conditions,
it should be considered novel;
4. A solution might generalize to a broader class of problems, while
another solution might be specific to certain conditions. In such
cases, they are considered distinct;
5. If the new solution is significantly simpler or more complex than
the others, it can be regarded as essentially novel, even if they lead
to the same result.

Given the following mathematical problem:
```

```
{problem}

Reference solutions:
{reference_solutions}

New solution:
{new_solution}

Please respond ONLY with a JSON object in this exact format:
{"verdict":"YES","error":""}
or
{"verdict":"NO","error":"<brief reason for the lack of novelty>"}

Output YES only if the new solution is novel compared to the reference
solutions. Do not include any extra text.
```

G.1.2 CS4 JUDGE TEMPLATES

Constraint-Satisfaction User Template (Verbatim).

```
You are an expert reader. I will give you a story followed by a set of
constraints. Your task is to carefully read both of them and tell how
many constraints are being satisfied in the story.

Judge every constraint independently and conservatively.
- Include a constraint number only if that constraint is fully
satisfied.
- Omit a constraint number if the constraint is not satisfied or only
partially satisfied.
- Constraint numbers are 1 through {num_constraints}.
- Do not include explanations, evidence, quotes, markdown, or extra
commentary.

Output exactly two lines:
SATISFIED: [comma-separated satisfied constraint numbers, or NONE]
COUNT: [number of satisfied constraints]

Story:
{story}

Constraints:
{constraints}
```

Quality/Relevance User Template (Verbatim).

```
You are an expert story evaluator. Evaluate the following story on
four dimensions using the anchor rules below.

Scoring anchors:
- 1/5 = severely deficient; major failures that make the dimension
largely unmet.
- 2/5 = weak; some partial evidence but clear and important problems
remain.
- 3/5 = adequate; the story is workable but has noticeable weaknesses.
- 4/5 = strong; the story meets the dimension well with only minor
weaknesses.
- 5/5 = excellent; the dimension is fully and clearly satisfied.

Dimension-specific guidance:
1. Grammar: judge sentence-level fluency, clarity, and absence of
obvious writing errors.
2. Coherence: judge whether the plot, causal flow, and scene logic
stay internally consistent from beginning to end.
3. Likability: judge reader-level engagement and readability, but do
```

```
not reward irrelevant ornament over clear story execution.
4. Relevance: judge how well the story actually uses the instruction
and constraints in its concrete content, not just surface keyword
overlap.

Instruction:
{instruction}

Constraints:
{constraints}

Story:
{story}

Output format:
Output exactly four lines and nothing else. In every line the score
MUST be a single Arabic digit from 1 to 5 (write 1, 2, 3, 4, or 5). Do
NOT use Roman numerals or tally marks such as "IIIII", do NOT use a
dash "-" in place of a number, and do NOT echo the placeholder text —
write a real digit and a real reason. Grammar: <digit 1-5>/5 - <brief
reason, under 12 words> Coherence: <digit 1-5>/5 - <brief reason,
under 12 words> Likability: <digit 1-5>/5 - <brief reason, under 12
words> Relevance: <digit 1-5>/5 - <brief reason, under 12 words>
```

Novelty User Template (Verbatim).

```
You are an expert evaluator. Rate how novel the story is given the
instruction and constraints.

Novelty anchors:
- 1/5 = highly conventional; mostly predictable, template-like, or
minimally changed from an obvious baseline.
- 2/5 = slightly novel; a few fresh details but the main story logic
remains generic or familiar.
- 3/5 = moderately novel; some meaningful fresh ideas, scenes, or
constraint integration, but still partly conventional.
- 4/5 = clearly novel; the story contains distinctive developments or
combinations that go beyond standard template responses.
- 5/5 = highly novel; the story is distinctly original in its core
developments while still respecting the task and constraints.

Judging guidance:
- Judge novelty by story content, event design, and constraint
integration.
- Do not reward incoherence, randomness, or gratuitous weirdness.
- Do not reward mere surface wording changes if the underlying story
is conventional.

Instruction:
{instruction}

Constraints:
{constraints}

Story:
{story}

Output format:
Output exactly one line and nothing else. The score MUST be a single
Arabic digit from 1 to 5 (write 1, 2, 3, 4, or 5). Novelty: <digit
1-5>/5 - <brief reason, under 12 words>
```

G.1.3 CFT Judge Templates

System Template.

```
You are scoring a Transformational Creativity benchmark answer.

Purpose:
- This is a system-reconstruction benchmark for disruptive
transformational creativity.
- In this variant, fluency and flexibility follow the psychology
creativity tradition, but the scoring is deliberately severe: fluency
means non-redundant ideational production; flexibility means
integrated breadth across genuinely different conceptual categories
within one answer.
- Novelty and appropriateness are scored independently. Novelty
concerns representational departure beyond the supplied rule change
and expected reconstruction; appropriateness concerns conditional
rule--goal fit, execution, verification, and failure handling.
- Do not treat fluency as language polish, length, completeness, or
number of bullet points. Do not treat flexibility as cross-item
robustness or simple checklist coverage.

Judging principles:
- Use only explicit evidence from the benchmark item and the model
answer.
- If a boolean condition is not clearly supported, mark it false. If
score evidence is ambiguous, choose the lower score.
- For fluency, count only distinct operational mechanism families. Do
not count substeps, repeated governance, monitoring, training,
rollout, fallback, audits, dashboards, logs, or renamed variants as
additional ideas unless they create a different causal mechanism for
the rebuilt rule world.
- For flexibility, count only integrated shifts across genuinely
different categories, such as representation, causal model,
institution, interface, record, incentive, validation logic, failure
semantics, stakeholder role, time horizon, and operational control.
Mere mentions do not count.
- A long, professional answer is normally not above 2.5 on fluency or
flexibility unless it contains non-redundant mechanism families and
cross-category interactions that change how the system works.
- Scores above 3.0 are reserved for answers that clearly exceed
ordinary comprehensive implementation plans. Scores above 4.0 should
be rare even among strong frontier-model answers.
- Score all four dimensions independently. Use task_completion,
goal_coverage, and rule_compatibility as diagnostic evidence, but do
not automatically zero all creativity scores when those diagnostics
are false. Do not let generic completeness inflate any score.

Output requirements:
- Return exactly one JSON object and nothing else.
- Do not output markdown, prose, <think> tags, or <answer> tags.
- The JSON object must satisfy the explicit JSON Schema in the user
prompt.
- Use the 0-5 score rubrics from the user prompt; half-point scores
such as 1.5, 2.5, and 3.5 are allowed.
- Do not add extra keys.
```

User Template (Instruction-Body Excerpt).

```
Benchmark item:
{item_json}

Model answer:
{answer_text}
```

```
Goal coverage keys to judge:
{goal_ids}

Active rule compatibility keys to judge:
{rule_ids}

Judging task:
- Score only explicit evidence in the model answer. If evidence is
ambiguous, choose the lower score.
- Completion diagnostics are evaluated first, but they are not hard
gates for the four creativity scores.
- task_completion=true only if the answer gives a final rebuilt
operating design, not just analysis, diagnosis, intentions, or a plan
to think further.
- goal_coverage=true for a goal only when the answer gives a concrete
mechanism plus an execution path or verification signal for that goal.
- rule_compatibility=true for a rule only when the answer states how
that rule changes behavior and names an enforcing mechanism, control,
interface, record, workflow, or decision rule.
- In higher-constraint items, mark rules false if the answer treats
constraints as a checklist instead of integrating their interactions.

Fluency rubric:
{fluency_rubric}

Novelty rubric:
{novelty_rubric}

Appropriateness rubric:
{appropriateness_rubric}

Flexibility rubric:
{flexibility_rubric}

Hard psychological calibration for fluency and flexibility:
- Fluency is non-redundant ideational quantity: count mechanism
families, not sentences, headings, bullets, components, or
implementation chores.
- Use this default anchor: an answer that is complete, practical, and
detailed but mostly an ordinary engineering plan should usually
receive fluency around 2.0-2.5, not 4.0.
- Fluency above 3.0 requires many non-redundant mechanism families
with concrete execution details. Fluency above 4.0 requires
exceptional ideational production with little generic padding.
- Flexibility is integrated breadth: categories only count when they
change design decisions and interact with each other under the
rewritten rule world.
- Do not reward checklist breadth. Stakeholders, records, validation,
migration, fallback, and governance are not separate flexibility
evidence if they are all routine consequences of the same operating
idea.
- Use this default anchor: a complete answer that touches many
implementation sections but does not change perspective or
representation should usually receive flexibility around 2.0-2.5.
- Flexibility above 3.0 requires coherent movement across different
conceptual categories and system layers. Flexibility above 4.0
requires unusual cross-category integration.

Independent calibration for novelty and appropriateness:
- Novelty measures incremental representational departure beyond the
supplied rule change and an expected reconstruction under that change.
- Appropriateness measures the conditional validity and operational
quality of the candidate reconstructed framework under the supplied
rules, goals, constraints, and execution environment.
```

```
- A conventional framework can receive high appropriateness when its
mechanisms are rule-compatible, traceable, executable, verifiable, and
failure-aware.
- A novel framework can receive low appropriateness when it violates a
supplied rule, leaves goals uncovered, or lacks a reliable execution
or validation path.
- If a response lacks a new representational primitive, state
variable, invariant, validity condition, measurement object, causal
explanation, unit of action, responsibility logic, or coordination
grammar, keep novelty at or below 2.0.
- Lower appropriateness according to explicit gaps in rule
compatibility, goal coverage, integration, execution, validation,
migration, or failure handling; unusualness and representational
departure do not raise this dimension.

Return exactly this JSON shape:
{json_shape}

JSON Schema:
{json_schema}

Output constraints:
- Return one JSON object only. Do not use markdown, prose, <think>, or
<answer> tags.
- Do not apply an automatic completion-gate zeroing rule. Set all four
scores to 0 only when the answer is unusable: empty, refusal, severe
off-task response, meaningless error text, or no reconstructive
solution attempt.
- If novelty_score >= 2.5, novelty_band_reason must explicitly name
the representational contribution beyond the supplied rule change and
identify at least two design parts or system layers it organizes.
- If appropriateness_score >= 2.5, evidence must explain which
mechanisms, records, interfaces, validation rules, decisions,
terminology, or institutions are traceable to the supplied rules and
goals, and how the proposed framework supports execution and
verification.
```

The four slots above are expanded with the complete half-point anchor lists in Appendix B.3. Thus the full executable prompt is the system template, the user instruction body, those four anchor lists, and the generated JSON shape/schema.

The reliability table in Appendix C.3 reports the resulting judge outputs at the task's first-level item granularity.

G.2 Adaptive Profiling and Workflow-Selection Prompts

The first prompt selects one Boden tier from the visible profile. This excerpt is copied from the tier classifier:

```
You are the routing layer of a creativity-solving system. Classify the
task into exactly ONE Boden creativity tier. Decide from the OUTPUT
SHAPE and the EVALUATION METRICS, NOT from the surface domain (a math
or coding task can still be exploratory).

Task profile (visible information only):
{_profile_block(task_name)}

Reason step by step in the "reason" field (output shape? which metric
family? domain is irrelevant), then give the tier. Return compact JSON
only:
{"reason": "output shape + metric family that decides the tier",
 "tier": "combinational|exploratory|transformational"}
```

After tier classification, the second prompt exposes only that tier's operator catalog and requests an ordered within-tier chain:

```
You are the routing layer. This task was classified as a {tier}
creativity task.
{objective_block}

Task profile:
{_profile_block(task_name)}
- canonical metric importance: {weight_str}
- HIGH-importance metrics that MUST be covered: {high_str}
{_sample_block(sample_text)}
You may ONLY choose from this {tier} operator pool (exact names | the
metrics each operator targets | what it does):
{catalog}

{procedure}

Rules:
- ORDERED list; execution order matters (extraction/mapping ->
generation -> verification/selection).
- Exact names from the pool above only. Do NOT invent or borrow
operators from other tiers.
- Do NOT include 'output_normalization' (appended automatically as the
final step).
```

G.3 Complete Baseline Prompt Catalog

This section lists the prompts used by the baseline methods. Braced blocks denote runtime insertion of the original task prompt or an earlier call's response.

Direct Prompting (Direct). Direct uses the original task prompt without an additional elicitation workflow.

Generic Creativity (Verbatim).

```
{base_prompt}

Apply a creativity bias. The most templated / textbook / first-thought
approach is OFF the table. Identify a less obvious angle: a surprising
rearrangement, an analogy from a far domain, a non-standard
decomposition, or a constraint reframing. State your creative angle
explicitly in one or two sentences before solving, then walk through
the reasoning that uses it. The reasoning must be visibly grounded in
the creative angle you chose.

After your reasoning, write the FINAL ANSWER between
<answer>...</answer> tags in the exact format requested by the task
above. The content inside <answer>...</answer> must be only the answer
(no extra commentary, no repeat of the reasoning). Whatever was inside
the tag will be parsed by the grader.
```

Structured CoT (Verbatim).

```
{base_prompt}

Apply chain-of-thought reasoning to the task above. Walk through the
problem step by step, numbering each step explicitly ("Step 1: ...",
"Step 2: ...", and so on). Be specific at each step about what you are
deciding and why. Do not skip the reasoning — the grader audits
whether your reasoning was actually shown.
```

```
After your reasoning, write the FINAL ANSWER between
<answer>...</answer> tags in the exact format requested by the task
above. The content inside <answer>...</answer> must be only the answer
(no extra commentary, no repeat of the reasoning). Whatever was inside
the tag will be parsed by the grader.
```

Self-Consistency (Verbatim Shared Prompt). Self-Consistency sends the Structured CoT prompt above three times at $\max(\textit{configured temperature}, 0.9)$, extracts each final answer, and programmatically majority-votes normalized answers; ties use first occurrence. The three generated outputs are the complete voter set.

Self-Refine, Draft Call (Verbatim).

```
{base_prompt}

This is the DRAFT phase of a self-refine workflow. Produce a
first-pass solution. Show your reasoning step by step.

After your reasoning, write the FINAL ANSWER between
<answer>...</answer> tags in the exact format requested by the task
above. The content inside <answer>...</answer> must be only the answer
(no extra commentary, no repeat of the reasoning). Whatever was inside
the tag will be parsed by the grader.
```

Self-Refine, Critique Call (Verbatim).

```
{base_prompt}

This is the CRITIQUE phase of a self-refine workflow. Below is your
DRAFT response. Critique it concretely:

- List specific constraint violations or factual errors
- List places the reasoning is shaky or missing a step
- List places the answer could be more creative / less templated /
more compliant with the requested format

Do NOT produce a new answer in this turn. Only output the critique as
a numbered list. Be honest and specific; if the draft is fine, say so
and explain why.

=== DRAFT ===
{draft_response}
=== END DRAFT ===
```

Self-Refine, Revise Call (Verbatim).

```
{base_prompt}

This is the REVISE phase of a self-refine workflow. Below are your
DRAFT and CRITIQUE. Now produce a revised solution that addresses
every concrete point in the critique. Show your reasoning step by step
(the revised reasoning, not just the diff).

=== DRAFT ===
{draft_response}
=== END DRAFT ===

=== CRITIQUE ===
{critique_response}
=== END CRITIQUE ===

After your reasoning, write the FINAL ANSWER between
```

```
<answer>...</answer> tags in the exact format requested by the task
above. The content inside <answer>...</answer> must be only the answer
(no extra commentary, no repeat of the reasoning). Whatever was inside
the tag will be parsed by the grader.
```

Single-Call ToT-Style Prompt (Verbatim).

```
{base_prompt}

Apply tree-of-thought search to the task above. Explore THREE distinct
solution branches, clearly labeled "Branch A:", "Branch B:", "Branch
C:". For each branch, state its core idea in one sentence, then
briefly reason about its viability and trade-offs (2-4 sentences).
After all three branches, write an EVALUATION section that compares
them against the task's constraints and picks one as the winner with a
clear justification.

After your reasoning, write the FINAL ANSWER between
<answer>...</answer> tags in the exact format requested by the task
above. The content inside <answer>...</answer> must be only the answer
(no extra commentary, no repeat of the reasoning). Whatever was inside
the tag will be parsed by the grader. The content inside
<answer>...</answer> is the winning branch's solution only — not the
other branches.
```

G.4 Operator Execution Reference

Every selected operator is executed with the item view and the preceding artifact. Appendix H.1 gives the shared execution template, an index of all 27 routable operators, and representative instruction bodies. The full instruction catalog and task-specific overlays are supplied with the code. Pool membership is specified in Appendix F.3.

H Operator Reference

H.1 Execution Template and Operator Index

Table 40 indexes all 27 routable operators by their role. The shared output-normalization step follows the selected workflow and is excluded from routing and ablation. The anonymous code supplement contains the full instruction bodies in `skills.py`, task-specific overlays in `task_prompts.py`, and the execution wrapper in `runtime_skills.py`.

Table 40: Complete routable operator index. Roles summarize the implementation; the full executable instruction text is provided in the code supplement.

Operator	Role
Combinational Pool	
`unit_extraction`	Identify reusable units
`relation_property_abstraction`	Abstract shared relations
`candidate_recombination`	Construct candidate combinations
`combination_verification`	Check candidate relations
`constraint_preservation`	Preserve stated constraints
`diversity_filtering`	Filter redundant combinations
`lexical_validity_check`	Check lexical validity
Exploratory Pool	
`constraint_space_mapping`	Map the feasible space
`exploration_axis_expansion`	Expand search strategies
`candidate_generation`	Generate distinct candidates

Continued on next page

Operator	Role
`coverage_balancing`	Balance candidate coverage
`novelty_transformation`	Develop alternative approaches
`semantic_deduplication`	Remove semantic duplicates
`feasibility_evaluation`	Check candidate feasibility
`execution_verification`	Verify code or mathematical steps
`coherence_check`	Check internal consistency
`portfolio_selection`	Select a valid candidate
Transformational Pool	
`rule_change_extraction`	Parse supplied rule changes
`legacy_assumption_mapping`	Map prior assumptions
`breakage_propagation`	Trace consequences of rule changes
`primitive_induction`	Introduce new representations
`system_reconstruction`	Rebuild the system architecture
`performance_reanchoring`	Restore performance under new rules
`norm_interface_establishment`	Define interfaces and standards
`residue_audit`	Check reliance on old assumptions
`goal_coverage_verification`	Verify operational goal coverage
`reconstruction_generation`	Write the complete reconstruction

Shared Execution Template (Verbatim Excerpt).

```
Task: {state.task_name}
Workflow: {state.workflow}
Skill: {self.name}
Instruction: {self.instruction}
Task-specific guidance: {task_guidance or
'Use the generic skill instruction.'}

Original prompt:
{state.original_prompt}

Visible item fields, excluding scoring-only fields:
{state.raw_item}

Previous step output:
{prev_step}

Return compact JSON. Include useful fields for this skill.
```

H.2 Representative Instruction Bodies

The following verbatim examples illustrate one operator from each pool. Selected operators use the shared template above; task-specific overlays are appended by the runtime.

`relation_property_abstraction`.

```
Abstract the relation, property, role, or semantic axis connecting the
extracted units before generating new combinations.
```

`candidate_generation`.

```
Generate multiple distinct candidates from different strategies before
selecting a final answer.
```

primitive_induction.

```
Introduce new primitives, variables, roles, interfaces, or validity
conditions required by the new rule world.
```