IDOL: ANSWER-FREE BEHAVIORAL CONSTRAINTS FOR SMALL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Behavioral constraints can guide a small language model without supplying a target response, but their contribution can be conflated with changes in prompting, candidate generation, and evaluation. We introduce IDOL, a source-labeled framework for measuring the incremental value of answer-free behavioral guidance. Typed IDOL-DSL programs distinguish requirements recoverable from the prompt from additional behavioral proposals, and compile into deterministic checks and semantic verifiers over student-generated responses. We evaluate canonical and prompt-derived programs through separate generation interventions and fixed-pool selector comparisons, using two automated judges in both response orders. On a fixed, preselected 300-prompt UltraChat-derived panel across three generation seeds, canonical guidance improves automated scores over a generic sham under both judges, while the closest explicit and permuted controls do not establish an advantage. Accepted latent constraints occur in 66 prompts and change 28 of 900 shared-pool selections relative to prompt-derived scoring. The targeted dual-order selector estimate is $+0.0239$ under DeepSeek, with a 95% interval of $[+0.0033, +0.0583]$, and $+0.0144$ under Phi, with $[-0.0067, +0.0500]$: both point estimates are positive, but the declared cross-judge support criterion is unmet. Controlled, filtered-rationale, and benchmark extensions yield inconclusive or mixed results. IDOL provides an executable representation and an attribution-focused study of constraint-guided inference, distinguishing a prompting gain from the incremental effect of additional scoring constraints.

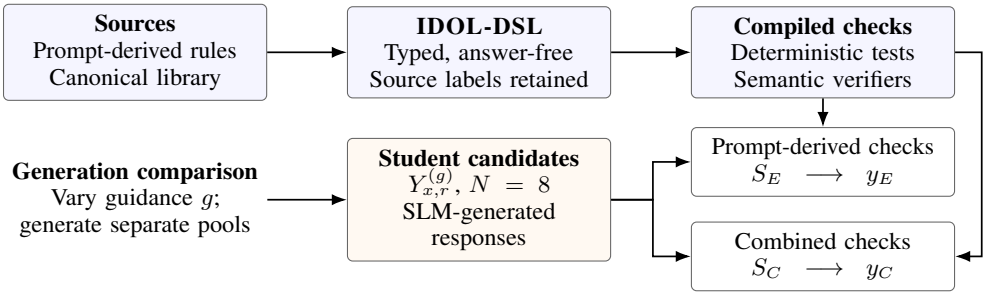

Figure 1: **Source-labeled constraints and matched evaluation.** Prompt-derived rules and a canonical library are represented as typed IDOL-DSL programs and compiled into declared checks. Generation interventions vary guidance and therefore the candidate pool. Selector comparisons reuse the same student-generated candidates, contrasting prompt-derived scoring with scoring that additionally includes accepted latent constraints.

1 INTRODUCTION

A behavioral instruction can change a language model's response without providing its answer: preserve the requested schema, separate checkable subproblems, or avoid unsupported certainty.

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.

Such guidance is attractive for small language models (SLMs), whose responses can be generated locally while requirements remain inspectable. Yet an improvement after adding guidance has several possible explanations. The prompt may elicit a different candidate pool; an explicit checker may select a better response; or an additional behavioral constraint may supply selection signal beyond requirements already recoverable from the prompt. These are different contributions and require different comparisons.

This paper asks: *when do additional behavioral constraints change an SLM's outputs beyond prompt-derived guidance, and how should that increment be measured?* We introduce IDOL, which represents answer-free guidance as source-labeled, executable constraint programs. Here *answer-free* means that the program specifies a behavioral requirement rather than a target response or item-specific solution. *Latent* is an operational source category for additional proposals not entailed by the prompt; it does not assert that a constraint is hidden teacher knowledge or independently valid. Every response candidate is generated by the student.

IDOL separates representation, generation, and selection (Figure 1). A compiler maps typed constraints to deterministic checks or semantic verifiers while retaining their source labels. Generation experiments vary the guidance supplied to the student. Selector experiments instead hold its candidate pool fixed and compare prompt-derived scoring against scoring with accepted latent constraints added. This design makes a stronger attribution test than comparing the whole pipeline only with a raw prompt, a generic caution, or a weak reranking baseline. Programmatic verification already supports instruction-following supervision in ACT and RLCF (**??**); our focus is the source-specific increment and the experimental separation needed to measure it.

The main study uses a fixed, post hoc panel of 300 UltraChat-derived prompts with synthetic profiles and three generation seeds. Canonical guidance improves automated evaluations over a generic sham under both evaluated judges, but comparisons with explicit and permuted guidance are inconclusive. The direct selector comparison identifies a different pattern: accepted latent constraints occur in 66 prompts, yet change only 28 of 900 prompt–seed selections. The targeted dual-order audit has positive point estimates under both judges; its interval-and-Holm rule is met under DeepSeek and not under Phi, so cross-judge support is unmet. A controlled panel, a filtered teacher-rationale panel, and objective benchmark extensions characterize the scope of these findings.

We make three contributions: **An inspectable representation and implementation.** IDOL-DSL encodes typed, source-labeled, answer-free constraints, compiles them to explicitly declared checks, and records their application to student outputs.

An attribution-focused evaluation design. We separate changes in generation from changes in fixed-pool selection, measure the increment beyond prompt-derived scoring, and evaluate controls with two judge families in both response orders.

An empirical characterization of constraint interventions. On the fixed public panel, a canonical-versus-sham gain coexists with a 3.11% selection-change rate and an unmet cross-judge criterion for the direct selector increment. We relate source coverage, actual selection changes, and outcome differences, alongside the controlled and benchmark results.

2 Related work

Distillation and behavioral supervision. Model compression and knowledge distillation transfer teacher outputs or probabilities (**??**). Reasoning traces can support prompting (**?**) or supervise smaller models (**?**). Instruction tuning and human feedback (**??**), preference optimization (**?**), and AI feedback (**??**) extend behavioral supervision. IDOL studies a different object: behavioral requirements applied at inference time, without teacher-written target responses or a student parameter update. This is a distinction in the supervision interface, not a claim that constraints outperform distillation.

Constraint verification and candidate selection. Outcome and process verifiers assess candidate solutions (**???**). ACT uses constraint satisfaction to construct preference supervision (**?**); RLCF uses instruction-specific checklists with programmatic and model-based verification (**?**). AutoIF builds instruction-following data through execution feedback (**?**); VerIF combines code and language-model verification (**?**). Posterior regularization constrains latent posteriors (**?**), while Snorkel combines noisy labeling functions (**?**). Accordingly, neither answer-free guidance nor verification alone is our

claimed distinction. IDOL combines source labels with direct comparisons against prompt-derived constraints, separating their generation and selection effects within an inspectable implementation.

Outcome measurement and selective intervention. RewardBench evaluates the behavior of learned preference models (**?**). Position and verbosity biases complicate model judging (**??**); our two-judge audit probes sensitivity rather than establishing human preference. Calibration, selective prediction, and uncertainty under distribution shift motivate several constraint families (**???**). However, predicting whether an additional scorer improves an existing response is different from predicting confidence in one response. We keep this distinction explicit when interpreting the routing experiment.

3 Problem formulation

Let x be a prompt and f_θ a fixed student. A source h provides an input-conditioned constraint set

$$\mathcal{C}(x) = \{(c_k, w_k, s_k)\}_{k=1}^{K}, \qquad w_k \in (0, 1], \tag{1}$$

where c_k is a checkable behavioral condition, w_k its weight, and s_k its source label. Write $\mathcal{C}_E(x)$ for requirements entailed by the prompt and $\mathcal{C}_L(x)$ for accepted additional epistemic, reasoning, scope, or structure constraints. These labels identify the intended source of a requirement; they do not certify its correctness or usefulness.

For generation condition g and seed r, the student samples $Y_{x,r}^{(g)} = \{y_i\}_{i=1}^{N}$. A compiled checker returns $v_k(x, y_i) \in [0, 1]$. For a nonempty accepted check set, the score is

$$S(x, y_i; \mathcal{C}) = \frac{\sum_{k \in \mathcal{C}} w_k v_k(x, y_i)}{\sum_{k \in \mathcal{C}} w_k}. \tag{2}$$

We use $S_E = S(\cdot; \mathcal{C}_E)$ and $S_C = S(\cdot; \mathcal{C}_E \cup \mathcal{C}_L)$ for prompt-derived and combined scoring. Their selected responses y_E and y_C maximize the respective scores on the *same* pool. When no latent constraint is accepted, combined scoring reduces to prompt-derived scoring. Unavailable checks and fallback decisions remain visible in the execution record.

The two experimental questions are now distinct. A generation comparison changes g and may change every response in $Y_{x,r}^{(g)}$. A selector comparison fixes x, r, g and the candidate strings, changing only the scorer. Its first observable is whether $y_C \neq y_E$; its second is the independently evaluated outcome difference. A larger internal checker score is not itself evidence that task utility or human preference improved.

4 The IDOL framework

4.1 Answer-free, source-labeled programs

IDOL-DSL represents each constraint with an identifier, type, source, weight, parameters, and checker target. Structural types cover schemas, requested sections, length bounds, keyword rules, and final-answer patterns. Epistemic types cover missing-evidence behavior, confidence language, and abstention. Reasoning types express checkable decomposition or consistency requirements; scope types restrict unsupported claims or extrapolation. The canonical library contains 68 constraints spanning four latent source categories. Appendix D gives the grammar, schema, and catalog.

The answer-free contract distinguishes a requirement from its solution. A program may request abstention when evidence is missing, but may not state whether the current item has missing evidence. It may require a JSON key, but may not fill that key with a target answer. Lexical scanners screen protected copied spans, answer markers, intermediate numerical solutions, and instruction echo. These executable guards do not establish semantic independence from the answer; they make the tested contract explicit (Appendix E).

4.2 Compilation, scoring, and decision records

The compiler resolves each DSL type to a deterministic checker or a semantic-verifier template. Parsing, schemas, regular expressions, length, and closed-label sets use deterministic checks. Semantic

Algorithm 1 Matched answer-free candidate selection

Require: Prompt x, source h, student f_θ, condition g, seed r, pool size N
1: Obtain an answer-free program $z \leftarrow h(x)$
2: Compile z into accepted prompt-derived and latent checks
3: Generate the student pool $Y_{x,r}^{(g)} = \{y_i\}_{i=1}^{N}$ once
4: Score every candidate using S_E and S_C on that same pool
5: Record both selected responses y_E, y_C and any fallback decisions
6: Bind program sources, checker outputs, and selected strings to their hashes
7: **return** y_E, y_C and the complete comparison record

checks ask a frozen evaluator whether a response follows the declared behavioral condition. Source labels are retained through compilation, so explicit-only, latent-only, and combined scores can be recomputed over the same responses.

A supported constraint is assigned a checker; an unsupported type is rejected; a proposal violating the answer-free contract is quarantined. Missing checker outputs are marked unavailable rather than silently interpreted as successful checks. Records retain the program, checker configuration, candidate hashes, individual scores, selector identity, and selected response. This connects source acceptance to observable decisions and permits comparisons to be reconstructed from response strings rather than historical candidate indices.

4.3 Generation and fixed-pool interventions

The generation factorial includes a raw prompt, explicit guidance, canonical discipline, a generic caution/checking sham, and a permuted control. These conditions can change the candidate distribution. Selection comparisons use shared pools: prompt-derived scoring uses checks obtained from the prompt; combined scoring adds accepted latent constraints. Thus "explicit" as a generation condition and "prompt-derived" as a scoring rule are not interchangeable (Table 1).

An exact-normalized response-mode baseline selects the most common normalized string. Its result depends on repetition and tie resolution within the pool. We retain this comparator as a diagnostic, but use the prompt-derived scorer for the central attribution test: changing selection relative to a weak baseline need not identify the contribution of additional constraints.

Table 1: Comparator map. Generation comparisons change guidance; selector comparisons fix candidate strings. Each public condition has $N = 8$ candidates per prompt–seed pool.

Contrast	Held fixed	Changed component
Canonical vs. sham	Prompts, seeds, sampling budget	Canonical guidance vs. generic caution/checking
Canonical vs. explicit / permuted	Prompts, seeds, sampling budget	Guidance condition; separate generated pools
Combined vs. prompt-derived	Same generated candidate pool	Prompt-derived checks vs. checks plus accepted latent constraints
Combined vs. exact mode	Same generated candidate pool	Combined scoring vs. normalized-response mode

Scope and extensions. Our experiments evaluate canonical and prompt-derived programs through generation interventions and fixed-pool selection. Teacher-authored executable programs and training from the resulting preference pairs are extensions of this interface, described in Appendix O. The routing experiment and revision timing measurements are reported separately from response-quality results. The LDTI diagnostic summarizes source coverage, distinction, candidate-score spread, checker availability, and verifier signal; it is not a calibrated probability of beneficial intervention.

5 EXPERIMENTAL DESIGN

5.1 RESEARCH QUESTIONS AND EVIDENCE PANELS

RQ1 asks how canonical guidance compares with explicit, sham, and permuted generation controls. **RQ2** asks how often accepted latent constraints change fixed-pool selection and whether those changes improve automated outcomes beyond prompt-derived scoring. **RQ3** examines the scope of these findings across evaluator families, response orders, controlled prompts, a filtered-rationale treatment, and objective benchmarks.

Fixed public panel. The primary panel contains 300 UltraChat-derived prompts (**?**), augmented with synthetic profiles, and seeds 13, 17, and 29, yielding 900 prompt–seed pools per selector condition. It was assembled post hoc and is fixed and preselected, not preregistered or randomly sampled from UltraChat. Of 300 source records, 261 produce accepted nonfallback objects and 66 contain accepted latent constraints. Eleven compiler inputs are rejected and 307 compiled constraints use semantic verifiers. Source coverage is measured over prompts; selection changes are measured over pools.

Controlled and filtered-rationale panels. A designed 32-prompt panel exercises explicit and latent behavioral conditions under the same three seeds. A separate 300-prompt factorial compares canonical, explicit, permuted, and filtered teacher-rationale conditions; 21 prompts satisfy the declared rationale filter. The latter treatment uses filtered strings, not complete executable teacher-authored programs. We report the full intention-to-treat panel and the filter-eligible subset separately (Appendix G).

Objective benchmark extensions. Frozen IFEval (**?**), IFBench (**?**), and GSM8K (**?**) runs compare blind-full selection with explicit-only selection. IFEval and IFBench use official strict prompt accuracy; GSM8K uses normalized exact match. Their analysis contract was frozen before outcome analysis. We retain full-set, subset, and cross-backbone configurations as distinct evaluations, rather than treating them as a single interchangeable benchmark population.

5.2 MODELS AND EVALUATORS

The headline student is `Qwen/Qwen3-0.6B` (**?**). For each prompt, condition, and seed it samples eight candidates at temperature 0.8, top-$p = 0.95$, and a 512-token cap with thinking disabled. Source labeling and semantic clustering use `Qwen/Qwen3-32B-AWQ`. Pairwise evaluators are `deepseek-ai/DeepSeek-R1-Distill-Qwen-7B` and `microsoft/Phi-3.5-mini-instruct` (**??**). Exact revisions and recovery settings appear in Appendix N.

Both judges run at temperature zero and evaluate every nonidentical pair in both response orders under the same blind rubric: directness, factual and internal soundness, usefulness and relevance, uncertainty handling, and requested format and scope, without rewarding verbosity alone. Exactly equal selected response strings are recorded as automatic ties, rather than assigned an inferred preference. All reported preference outcomes are automated; no human judgments were collected. Appendix M retains the annotation protocol and clearly separates its blank forms from outcome evidence.

5.3 ESTIMANDS AND STATISTICAL ANALYSIS

A public-panel contrast is indexed by generation condition, scoring rule, evaluator, and response-order analysis. Let $d_{x,r}^{(J)}$ denote the recorded paired judge-score difference for evaluator J, with positive values favoring the named intervention. Each valid orientation supplies integer response scores in $\{1, \ldots, 5\}$. For a nonidentical pair, we map A/B positions back to method and baseline identity and average the two orientation-specific method-minus-baseline score differences; for the direct selector contrast, this is combined minus prompt-derived. Exactly equal selected strings receive automatic 3–3 ties and zero differences in both orientations, without model inference. Directionally conflicting records are unresolved for winner-based analyses, but the primary score estimand retains their numerical differences when both orientations parse; a separate order-consistent sensitivity

Table 2: Headline automated contrasts on the fixed public panel. Two-way 95% intervals describe mean dual-order differences of 1–5 scores, not win rates. Generation rows are broad-factorial estimates; selector rows are from the targeted audit and differ from the broad DeepSeek estimate of $+0.0261$. Closest generation controls report the primary judge.

Contrast	Judge	Delta	95% interval
Canonical vs. generic sham	DeepSeek	$+0.1794$	$[+0.0422, +0.3150]$
Canonical vs. generic sham	Phi	$+0.2356$	$[+0.0933, +0.3850]$
Canonical vs. explicit	DeepSeek	$+0.0128$	$[-0.0239, +0.0489]$
Canonical vs. permuted	DeepSeek	$+0.0006$	$[-0.0544, +0.0572]$
Combined vs. prompt-derived (targeted)	DeepSeek	$+0.0239$	$[+0.0033, +0.0583]$
Combined vs. prompt-derived (targeted)	Phi	$+0.0144$	$[-0.0067, +0.0500]$

excludes them. All 900 reported direct-selector pairs have two valid parses. We average the three seed-level paired differences within each prompt and then average the 300 prompt means; because the panel is complete and balanced, this also equals the mean over all 900 pools. The estimate is *not* the average over only the 28 changed selections. Judge-score deltas are not win percentages or changes in benchmark accuracy. The separately reported objective benchmark deltas are paired differences on their declared accuracy metrics.

For the three-seed panels, we average paired differences across seeds within prompt. Two-way 95% percentile-bootstrap intervals independently resample prompts and seeds with replacement for 20,000 replicates. Two-sided paired sign-flip tests use seed-averaged prompt effects and 100,000 resamples, with Holm correction (**?**) within each declared contrast family. In the broad factorial analysis, a within-judge robust-positive result must satisfy the declared interval and corrected-test criteria in both the primary and response-order-consistent analyses. The targeted selector audit applies its declared interval-and-Holm rule to the dual-order score aggregate and is reported alongside, not substituted for, the broad order-consistent sensitivity. This targeted selector family is a post-inspection corrective audit; its within-family Holm adjustment does not remove that analysis-stage adaptivity. Cross-judge support additionally requires both evaluators. Only three seeds make seed-variability inference low-resolution; resampling the fixed panel does not establish representativeness of a deployment population.

An order-consistent preference requires compatible decisions after reversing response order and mapping decisions back to the same responses. We distinguish identical responses from judged ties, conflicting directions, and unparseable judgments; unresolved cases do not establish a directional preference even when valid scores allow them to remain in the primary score-difference estimand. Appendix C records the analysis contract. Differences in whether two judges meet a support threshold are not themselves a test of whether their effects differ (**?**).

6 Results

6.1 Canonical guidance and its closest generation controls

Canonical guidance improves automated evaluations relative to a generic caution/checking sham on the fixed public panel. DeepSeek gives $+0.1794$ with a two-way 95% interval $[+0.0422, +0.3150]$; Phi gives $+0.2356$ with $[+0.0933, +0.3850]$. This positive result is visible under both reported evaluator families and supports a generation-condition effect.

The closest controls qualify its attribution. Under the primary DeepSeek analysis, canonical guidance versus explicit guidance yields $+0.0128$ with $[-0.0239, +0.0489]$; versus permuted guidance it yields $+0.0006$ with $[-0.0544, +0.0572]$. Neither establishes an advantage (Table 2). Moreover, 234 of the 300 prompts have no accepted latent source, and the latent-source subgroup does not retain a positive result after the declared order and multiplicity checks. Thus the sham comparison supports the usefulness of this canonical guidance condition relative to that control, not the incremental value of its latent component.

6.2 SELECTION REACH AND THE INCREMENTAL SCORING EFFECT

Adding accepted latent constraints changes 28 of 900 shared-pool selections (3.11%); the remaining 872 (96.89%) select the same response string. This census measures the *operational reach* of the additional scorer, not the quality of its changes. Accepted latent-source coverage, 66 of 300 prompts (22.00%), is a different quantity: a source can be accepted without altering which candidate maximizes the score (Table 3).

Across all 900 pools, the targeted dual-order selector estimate is $+0.0239$ under DeepSeek, with $[+0.0033, +0.0583]$, and $+0.0144$ under Phi, with $[-0.0067, +0.0500]$. The targeted rule is met under DeepSeek but not Phi, so cross-judge support is unmet. A separate broad DeepSeek run gives $+0.0261$ and an order-consistent sensitivity of $+0.0250$; Appendix G reconciles the deliveries. The evidence is therefore a positive but incompletely corroborated automated signal, not a demonstrated between-judge difference.

Table 3: Source coverage and decision changes have different denominators. The direct selector census covers all 900 prompt–seed pools. The exact-mode rows concern a separate baseline diagnostic.

Quantity	Count / denominator	Rate
Prompts with an accepted latent source	66/300	22.00%
Combined = prompt-derived response	872/900	96.89%
Combined $\neq$ prompt-derived response	28/900	3.11%
Exact mode selects candidate 0	841/900	93.44%
Combined $\neq$ exact-mode response	341/900	37.89%

Why the baseline matters. The combined-versus-exact-mode comparison changes 341 selections, but exact mode selects candidate 0 in 841 pools. Its disagreements are longer on average, length correlates with both judge scores, and the signed improvement is concentrated in one profile (Appendix H). These observational patterns neither adjust causally for length nor apply to the 28-pool latent comparison; the weak baseline cannot replace the prompt-derived attribution test.

6.3 CONTROLLED AND FILTERED-RATIONALE INTERVENTIONS

The 32-prompt controlled panel yields inconclusive contrasts. Its primary latent-versus-explicit generation estimate is $+0.1302$, with $[-0.0938, +0.3906]$; the combined-versus-explicit selector estimate is -0.0156, with $[-0.1875, +0.1302]$. No declared contrast meets the robust positive criterion. The wide intervals do not establish equivalence, nor does the negative selector point estimate establish harm.

The tested filtered-rationale intervention also provides no robust evidence of improvement under the declared order-consistent and multiplicity-adjusted analyses, either in the full 300-prompt panel or the 21-prompt eligible subset. Several eligible-subset point estimates are negative under both judges. This concerns the tested filtered strings, not rationale prompting generally or executable teacher-authored programs; Appendix G retains the estimates.

6.4 EVALUATOR AND RESPONSE-ORDER SENSITIVITY

Across the full active corpus, 35,262 of 48,912 pair records compare identical responses. Of 13,650 nonidentical pairs, 5,254 are conflicting or unresolved across response order (38.49%). This whole-corpus quantity combines conflicts and unresolved judgments; it is neither a pure reversal rate, a human-ground- truth error rate, nor a rate for the 28 changed selector pairs.

Dual-order evaluation exposes instability hidden by a single orientation. Canonical versus sham meets the declared criterion under both judges; the selector increment does not meet the cross-judge criterion. Different threshold status is not a formal between-judge effect comparison.

6.5 Objective benchmark outcomes

Objective accuracy provides an outcome measure separate from automated preference (Table 4). IFEval and IFBench contrasts span zero. On GSM8K, the full 1,319-item, seed-13 Qwen3-0.6B run gives $+0.0273$, with $[+0.0114, +0.0432]$. The separate three-seed, 300-item panel gives $+0.0111$, with $[-0.0100, +0.0333]$; it does not corroborate the full-run result, and their population and seed coverage differ. Both cross-backbone subset intervals also span zero. The outcomes are mixed and configuration-specific.

Table 4: Objective outcomes: blind-full minus explicit-only selection. Deltas are accuracy differences. IFEval and IFBench use strict prompt accuracy; GSM8K uses normalized exact match. Full and hash-300 GSM8K panels differ in population and seed coverage.

Benchmark	Student / evaluation panel	n	Delta	95% interval
IFEval	Qwen3-0.6B, full	541	$+0.0111$	$[-0.0296, +0.0518]$
IFBench	Qwen3-0.6B, full	300	-0.0100	$[-0.0333, +0.0100]$
GSM8K	Qwen3-0.6B, full, seed 13	1,319	$+0.0273$	$[+0.0114, +0.0432]$
GSM8K	Qwen3-0.6B, hash-300, three seeds	300	$+0.0111$	$[-0.0100, +0.0333]$
GSM8K	Qwen2.5-1.5B, hash-300	300	-0.0233	$[-0.0700, +0.0233]$
GSM8K	Llama-3.2-3B, hash-300	300	$+0.0133$	$[-0.0267, +0.0533]$

6.6 Routing, resource cost, and execution integrity

A ridge gate trained on 841 IFBench/IFEval records selects blind IDOL in 91.91% of cross-validated cases and realizes a mean gain of $+0.0048$. Nearby penalties select it in 91.44% of cases, with $+0.0059$ gain. The gate is nearly always on, and LDTI sweeps identify no universal threshold (Appendix J).

The latency-only $N = 16$ microbenchmark is separate from the scientific $N = 8$ panels. On four A100s, cached selection is $1.1586\times$ explicit best-of-16 and $2.1599\times$ $N = 1$; revision raises the ratios to $1.2151\times$ and $2.2653\times$. These are model-resident stage sums, not service latency or evidence of revision quality (Appendix K).

The execution record traces 70,560 candidates, 67,320 selections, 48,912 pairs, and 97,824 orientations without a reported provenance defect. Two locked regenerations reproduce 14 statistical files and 10,489 numerical fields exactly (Appendix L).

7 Discussion

Separating prompting gains from scoring gains. A canonical-guidance improvement over a generic sham and a low rate of latent-specific selection changes are compatible findings. The former changes what the student generates; the latter asks whether adding a scorer changes the choice among already generated candidates. Combining these comparisons under a single "constraint benefit" would obscure where the observed effect enters. The contribution of IDOL is to make that distinction executable and measurable through source labels and shared pools.

Decision frequency and conditional benefit. For task utility U and combined and prompt-derived choices y_C, y_E, let $\Delta U = U(x, y_C) - U(x, y_E)$. Since identical responses have zero utility difference, the law of total expectation gives

$$\mathbb{E}[\Delta U] = \Pr(y_C \neq y_E)\, \mathbb{E}[\Delta U \mid y_C \neq y_E]. \tag{3}$$

The observed first factor is $28/900$, or 3.11%. This separates how often an additional constraint changes the decision from how beneficial those changes are. A low change rate alone establishes neither uselessness nor safe preservation of a good baseline. Likewise, preference among changed pairs is not the whole-panel effect. Automated preferences require external validation before they can be interpreted as human benefit. For a specified task utility whose range has width B, the identity also implies $|\mathbb{E}[\Delta U]| \leq B \Pr(y_C \neq y_E)$. This is a scale-dependent bound on the possible aggregate

effect, not an estimate of improvement. Increasing the number of judgments on changed pairs can improve measurement of their conditional effect, but cannot expand the intervention's observed reach. Both the full-panel denominator and the changed-pair denominator are therefore necessary for interpreting a selector.

From accepted sources to useful interventions. Acceptance, candidate discrimination, and outcome improvement are separate stages. A valid program can assign every candidate the same latent score or change scores without changing the maximizing response. The present source coverage and decision census quantify two of these stages, but do not identify the causes of every unchanged selection. A prospective evaluation can test whether source applicability, genuine candidate diversity, and independently validated checker signal predict a beneficial intervention. That is a testable routing hypothesis, not a policy validated by the current near-always-on gate.

The role of candidate diversity. Fixed-pool attribution is conditional on the student and sampling budget: constraints cannot select a response that the student never generated. Increasing N changes both the opportunities for selection and the cost of checking candidates. The observed change rate therefore characterizes this student, these guidance conditions, and $N = 8$, rather than an intrinsic property of latent constraints. Comparisons involving new sources or samplers should match candidate pools or explicitly identify the generation change.

Modular extensions. Separating the representation from its source allows future studies to compare source-generation quality while holding constraint application fixed. Teacher-authored programs need to satisfy the same answer-free and checker contracts; their benefit does not follow from the canonical results. Recorded scores can also define preference pairs, but learning from them requires a separate test of transfer to task outcomes. These extensions build on an implemented inference study rather than supplying evidence for its current conclusions.

8 LIMITATIONS

The public panel is fixed, post hoc, and paired with synthetic profiles; its intervals describe the recorded panel and seed resampling, not random sampling from a deployment population. The controlled panel is small, and only 21 rationale prompts pass the filter. All preference outcomes are automated; source-label relevance, semantic leakage, verifier calibration, and human benefit lack independent human validation. The corpus-wide response-order census does not establish the corresponding rate for the 28 changed selector pairs. Objective benchmark effects vary by configuration, and neither the nearly always-on gate nor model-resident timing validates a deployment policy. Checker implementation burden and annotation time are unmeasured. Teacher-program transfer, preference training, and distilled verification require separate evaluations.

9 CONCLUSION

IDOL makes answer-free behavioral guidance inspectable through source-labeled programs, declared checkers, and student-generated candidate pools. Its matched design distinguishes generation-condition effects from the increment of additional scoring constraints. On the fixed public panel, canonical guidance improves automated evaluations over a generic sham, while the direct latent-selector comparison changes 28 of 900 selections and does not meet the declared cross-judge support criterion. These findings motivate evaluating constraint systems through source coverage, actual decision changes, and separately assessed outcomes. The result is an executable representation and an attribution-focused study of constraint-guided inference, with broader transfer and human benefit requiring independent validation.

AI USE STATEMENT

In this work, we used generative AI tools for feedback on research methodology and experimental protocols, method implementation, consistency and lineage checks, and interpretation of results. Within the experimental pipeline, language models generated synthetic response datasets (student candidates and teacher-rationale strings), supported constraint-source labeling and semantic clustering, and produced semantic-verifier scores and automated pairwise judgments (Sections 4 and 5; Appendix N). The teacher-rationale panel uses filtered strings, not complete executable teacher-authored programs.

Additionally, generative AI tools assisted with literature retrieval, identification and organization of related work, reference checking and formatting, software and research-artifact preparation, and drafting and editing manuscript sections for structure, consistency, and readability.

All AI-assisted work was reviewed by an author, who checked model and data provenance, implementation and experiment outputs, statistical analyses, citations, figures, and manuscript claims. Implementation and analysis checks include the tests, response-string fingerprint and equality checks, and locked regenerations reported in Appendix L. These checks establish execution and analysis consistency within their stated scope; they do not constitute independent human validation of model-generated labels, verifier scores, or judgments. No human judgments were collected. We take responsibility for the final content of this work, including text, code, analyses, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Code and research artifacts for anonymous review are provided at `https://anonymous.4open.science/r/IDOL-ICLR27-8B0A`. Appendix L describes the artifact contents and verification checks.

Sections 3 and 4, together with Algorithm 1, specify the constraint representation, scoring rules, and candidate-selection procedure. Table 1 distinguishes generation interventions from fixed-pool selector comparisons. Section 5 describes the evidence panels, generation seeds, model and decoding settings, evaluation rubric, response-order protocol, and statistical analysis, and the distinction between fixed-pool and generation-level estimands. Section 6 and Appendix L report candidate and selected-response provenance, response-string fingerprint checks, and pairwise-comparison integrity. Sections 7 and 8 distinguish reproducibility of the executed pipeline and its automated analyses from semantic validity, human benefit, and generalization.

ETHICS STATEMENT

The reported evaluation uses public benchmark data, a fixed UltraChat-derived panel with synthetic profiles, and a designed controlled panel. No completed human judgments were collected; synthetic profiles and automated evaluations are not treated as evidence of real users' preferences. Answer-free constraints do not guarantee factual correctness or semantic independence from task answers, and their source labels and semantic verifiers have not been validated through human annotation. Behavioral constraints may encode normative assumptions about appropriate uncertainty, abstention, and scope, while automated judges can favor responses for reasons unrelated to human benefit. The automated outcomes and observed response-order sensitivity do not justify interpreting score improvements as evidence of safety, fairness, or alignment with human preferences. Accordingly, IDOL is presented as an experimental framework rather than a validated deployment policy. Use beyond this research setting would require independent validation of constraint appropriateness and verifier reliability, task-grounded evaluation, and appropriate human oversight. Relevant empirical boundaries are discussed in Sections 7 and 8.

REFERENCES

APPENDIX

A APPENDIX GUIDE

The appendix serves four purposes. Sections B– C define the evidence panels and statistical decision rules. Sections D–F specify the transfer object and its implementation. Sections G– K expand the positive, null, and adverse results. The final sections document reproducibility, human-evaluation status, model and data scope, extensions of the source and training interfaces, and broader impacts. Counts and effects reported here refer to the response strings bound to the analyses.

Table 5: Evidence map. Each conclusion is scoped to its measured contrast.

Question	Observed evidence	Supported conclusion
Public intervention	Canonical > sham under both judges	Generation effect, not a latent-specific increment
Minimal latent selector	28/900 decisions differ; both point estimates positive	Cross-judge criterion unmet
Controlled mechanism	Wide intervals; no robust positive contrast	Benefit inconclusive
Teacher rationale	No robust positive contrast	Tested filtered-string benefit inconclusive
Full public benchmarks	Mixed; one full-set single-seed GSM8K interval positive	Broad gain not established
Learned routing	91.91% selection; $+0.0048$ gain	Gate is nearly always-on
Human validity	Zero completed judgments	Human benefit unknown

B STUDY REGISTRY AND EVIDENCE HIERARCHY

Table 6 records the role of each panel. The hierarchy is important because the panels answer different questions. The public panel has the largest fresh factorial but is fixed and preselected. The controlled panel isolates interventions but is small. The rationale panel evaluates one specific prompting implementation rather than the full DSL. Frozen benchmarks broaden task coverage and yield configuration-specific positive and inconclusive estimates.

Table 6: Study registry. A prompt count is not a claim of population representativeness.

Panel	Prompts	Seeds	Evidential role
Fixed public	300	13, 17, 29	Generation factorial and fixed-pool selector tests
Controlled	32	13, 17, 29	Designed mechanism and control panel
Teacher rationale	300	13, 17, 29	Filtered rationale-string intervention; 21 eligible
IFEval	full frozen run	frozen	Natural instruction-following boundary
IFBench	full frozen run	frozen	Verifiable instruction-following boundary
GSM8K	full frozen run	frozen	Cross-domain arithmetic boundary

Post hoc status. The 300-prompt post hoc panel was assembled after the original study and is not preregistered. Its fixed slice, conditions, and multiplicity families are fully recorded, but the results should be interpreted as exploratory evidence with strong lineage controls rather than as a confirmatory population study.

Source coverage. The public panel contains 300 source records. Of these, 261 produce accepted nonfallback source objects and 66 contain at least one accepted latent source. The canonical latent library contains 68 constraints spanning four source categories. Eleven compiler inputs are rejected, and 307 compiled items use semantic-verifier checks. Teacher-authored free-form constraint text is not used as the canonical downstream intervention; this matters when interpreting the canonical-versus-sham contrast.

C STATISTICAL DECISION RULES

C.1 PAIR CONSTRUCTION AND EQUALITY

Every comparison is keyed by study, prompt identifier, generation seed, condition, selector, judge family, and response orientation. Equality is computed by exact comparison of the current selected response strings. A comparison whose two strings are exactly equal is assigned an automatic 3–3 tie in both orientations and contributes a zero difference to the direct selector aggregate rather than an inferred preference. The delivered census contains no normalized-equal but textually different direct-selector pair, so normalization would not change the reported 872 equal and 28 nonidentical counts. This rule prevents historical candidate indices from being mistaken for current response differences after regeneration or reselection.

C.2 RESPONSE-ORDER RULE

For a nonidentical pair (a, b), each judge evaluates both (a, b) and (b, a). Each valid orientation yields integer response scores in $\{1, \ldots, 5\}$. The parser rounds a numeric score to an integer, rejects scores outside that range, and requires the winner label to agree with the score ordering. After mapping A/B positions back to selector identity, the pair-level score difference is the mean of the two orientation-specific method-minus-baseline differences and therefore lies in $[-4, +4]$ in half-point increments. Judged ties, conflicting mapped winners, and unparseable judgments are distinct from exact response equality. Conflicting mapped winners remain unresolved as directional preferences rather than being broken by a hidden heuristic. The primary score estimand nevertheless retains an unresolved pair when both score records parse; the separate order-consistent sensitivity excludes unresolved pairs. All 900 pairs in the reported direct-selector panel have two valid parses. Three seed-specific differences are averaged within each prompt, followed by an average over the 300 prompt means; for this complete balanced panel, that equals the mean over all 900 pairs. Section I gives the corpus-wide winner-agreement census, whose denominator is not the 28 changed selector pairs.

C.3 INTERVALS AND MULTIPLICITY

For each declared contrast, we average paired score differences over the three generation seeds within prompt. We report two-way 95% percentile-bootstrap intervals formed by independently resampling prompts and generation seeds with replacement for 20,000 replicates. Separately, we compute a two-sided paired sign-flip test on seed-averaged prompt effects using 100,000 resamples and apply Holm correction (**?**) within each declared contrast family. In the broad factorial analysis, a within-judge result is called robust-positive only if both the primary and response-order-consistent analyses meet the prespecified interval and corrected-test criteria. The targeted selector audit applies its declared interval-and-Holm rule to the dual-order score aggregate; its direct estimate is reported alongside, not substituted for, the broad matrix's order-consistent sensitivity. Cross-judge support additionally requires both judge families. With only three generation seeds, inference over seed variability remains low-resolution. Separate row/column resampling is studied by **?**; this methodological precedent does not establish coverage for our small seed dimension or exploratory panel. The multiplicity procedure follows **?**, without correcting analysis-stage adaptivity.

C.4 EXACT REGENERATION

Two complete regenerations use locked analysis-source and raw-pair hashes. They compare 14 output files and 10,489 numerical fields, with no numerical or non-runtime payload mismatch. This test covers determinism of the delivered analysis. It does not convert an automated preference into an objective quality label.

D IDOL-DSL SPECIFICATION

D.1 DESIGN GOALS

IDOL-DSL is designed to keep the constraint-source contribution compact, answer-free, source-aware, and executable. Each constraint must state a behavioral requirement rather than a target answer; declare whether it follows from the prompt or is proposed as latent discipline; provide the parameters needed by a checker; and avoid dependence on hidden task labels. The compiler rejects rather than repairs an unsupported type. The reported panels populate this representation with canonical and prompt-derived constraints. The source interface is independent of the candidate generator and checker stack.

D.2 INFORMAL GRAMMAR

```
program      := { version, constraints[] }
constraint   := { id, type, source, weight, parameters, checker }
source       := explicit
              | latent_epistemic
              | latent_reasoning
              | latent_scope
              | latent_structure
checker      := deterministic(name, parameters)
              | semantic(rubric, scale)
weight       := real in (0, 1]
```

The grammar deliberately excludes fields for an expected response, target label, chain of thought, or item-specific solution. Task adapters may add checker parameters, but not teacher-authored answer content.

D.3 CANONICAL SCHEMA

```
{
  "version": "idol-dsl-1",
  "constraints": [
    {
      "id": "c1",
      "type": "missing_evidence_abstention",
      "source": "latent_epistemic",
      "weight": 0.8,
      "parameters": {
        "allowed_abstention_markers": ["insufficient information"]
      },
      "checker": {
        "kind": "semantic",
        "rubric": "Does the response avoid unsupported claims?",
        "scale": [0, 1]
      }
    }
  ]
}
```

This example illustrates the representation, not a validated task outcome. The semantic rubric still requires calibration before its score can be treated as a useful outcome.

Table 7: Core IDOL-DSL families and their intended compilation targets.

Family	Typical source	Behavioral intent	Checker class
Schema/format	Explicit	Preserve requested keys, labels, or layout	Parser or regex
Length/content rule	Explicit	Respect requested bounds and inclusions	Token/string check
Missing evidence	Latent epistemic	Avoid unsupported completion	Semantic plus markers
Confidence calibration	Latent epistemic	Match certainty language to evidence	Semantic rubric
Decomposition	Latent reasoning	Separate checkable subproblems	Structural/semantic
Consistency	Latent reasoning	Avoid incompatible intermediate claims	Semantic rubric
Scope control	Latent scope	Stay within supplied evidence or role	Semantic rubric
Safety boundary	Explicit or latent scope	Refuse or redirect prohibited content	Policy checker
Response organization	Latent structure	Improve readable organization when useful	Structural rubric

D.4 Constraint catalog

E Prompt contracts and leakage guards

E.1 Teacher program prompt: source-interface extension

The teacher-program extension, not used as the headline downstream intervention, has four essential instruction clauses. It requests behavioral constraints only; prohibits answers, calculations, copied facts, and item-specific labels; requires a source label for every constraint; and asks for strict JSON matching IDOL-DSL. It also instructs the teacher to return an empty latent set when no latent norm is justified. This final option is necessary because forcing a latent proposal would turn coverage into an artifact of the prompt.

```
Read the user request and describe only checkable behavioral requirements.

Do not answer the request. Do not provide a solution, label, fact,
    numeric
result, rationale, or intermediate value that would help solve it.

Label each requirement as EXPLICIT if it is entailed by the request;
otherwise use one of the declared LATENT source labels. A latent
    requirement
must state a general behavioral norm, not item-specific content. If no
    latent
norm is justified, return no latent requirement.

Return valid IDOL-DSL JSON only.
```

E.2 Deterministic screening

The screening layer checks schema validity, unsupported keys, response-like answer markers, copied n-grams from protected task fields, explicit numerical solutions, and several forms of generic instruction echo. Two previously missed echo families motivated a scanner expansion and complete regeneration of the two affected eight-candidate pools. Both declared scanners report zero final lexical findings in generated and selected outputs. This statement is strictly lexical: semantic contamination remains possible and unmeasured.

E.3 SOURCE-LABEL DISTINCTION

An explicit constraint must be recoverable from the user request without consulting teacher behavior. A latent constraint proposes a norm that the user did not request but that may reduce undesirable behavior. Because this classification itself can be subjective, human source-label validation is a necessary next step. The current artifact records labels but does not contain completed human assessments of their accuracy.

F COMPILER AND VERIFIER IMPLEMENTATION

F.1 COMPILATION PIPELINE

The compiler performs six stages:

1. parse strict JSON and validate the declared DSL version;
2. reject missing, duplicated, or unsupported constraint identifiers;
3. run answer-leakage and instruction-echo screens;
4. resolve deterministic types to named checker implementations;
5. render semantic types into frozen, source-recorded verifier prompts;
6. emit an immutable compiled record with hashes for the source program, checker configuration, and prompt.

An execution record stores the student candidate hash, each checker output, the weighted score, the selected candidate hash, and the selector identity. This level of bookkeeping is what allows equality and pair lineage to be recomputed from delivered responses.

F.2 DETERMINISTIC AND SEMANTIC CHECKS

Deterministic checks are preferred whenever a requirement can be evaluated by parsing or exact computation. Semantic verification is reserved for nonlocal properties such as unsupported certainty or scope violations. A semantic score is bounded in $[0, 1]$ and is never silently substituted for a missing deterministic result. If a declared checker cannot run, the constraint is marked unavailable and the decision record exposes the fallback.

F.3 SCORE DECOMPOSITION

For candidate y, let E and L index explicit and accepted latent constraints. For nonempty source subsets, the decomposed scores are

$$S_E(x, y) = \frac{\sum_{k \in E} w_k v_k(x, y)}{\sum_{k \in E} w_k}, \tag{4}$$

$$S_L(x, y) = \frac{\sum_{k \in L} w_k v_k(x, y)}{\sum_{k \in L} w_k}, \tag{5}$$

$$S_C(x, y) = \frac{\sum_{k \in E \cup L} w_k v_k(x, y)}{\sum_{k \in E \cup L} w_k}. \tag{6}$$

When L is empty, the latent-only normalized expression is not used and combined scoring reduces to prompt-derived scoring; unavailable-check and empty-set fallbacks remain recorded rather than interpreted as positive evidence. The final equality contract verifies this behavior over current strings and scores.

F.4 WHEN ADDED SCORES LEAVE SELECTION UNCHANGED

Let $W_E = \sum_{k \in E} w_k$ and $W_L = \sum_{k \in L} w_k$. For fixed available-check sets with positive total weight in both sources,

$$S_C(x, y) = \frac{W_E}{W_E + W_L} S_E(x, y) + \frac{W_L}{W_E + W_L} S_L(x, y). \tag{7}$$

If S_L is constant over the pool, the prompt-derived ranking is unchanged. A nonconstant latent score can also preserve the same maximizing response. With a common deterministic tie rule, either case can leave the selected string unchanged; different candidate indices can also normalize to the same string. These are analytical possibilities, not a measured partition of the 872 unchanged selections. Quantifying them requires the per-pool scores, availability masks, tie records, and normalized strings. In particular, the 66-prompt source census cannot by itself supply that pool-level partition.

F.5 Candidate-pool reuse

The minimal selector comparison reuses the same generated pool across scoring rules. This prevents generation randomness from being attributed to the selector. Generation interventions are analyzed separately because their prompts may change every candidate. Pool reuse metadata identify whether a candidate set is shared, regenerated, or unavailable; historical pre-reuse payloads that cannot be independently reconstructed are not represented as fresh evidence.

G Full automated result summary

G.1 Public primary-judge contrasts

Table 8 expands the main public results. A status of "not established" means that the declared interval or corrected test does not support a positive conclusion. It does not imply equivalence.

Table 8: Selected public-panel contrasts under the primary DeepSeek analysis.

Contrast	Mean delta	Two-way 95% interval	Status
Generation: canonical vs. explicit	$+0.0128$	$[-0.0239, +0.0489]$	Not established
Generation: canonical vs. sham	$+0.1794$	$[+0.0422, +0.3150]$	Positive within judge
Generation: canonical vs. permuted	$+0.0006$	$[-0.0544, +0.0572]$	Not established
Total combined vs. explicit	$+0.0283$	$[-0.0083, +0.0667]$	Not established
Raw selector combined vs. explicit	$+0.0089$	$[-0.0022, +0.0300]$	Not established
Explicit selector combined vs. explicit	$+0.0144$	$[-0.0028, +0.0500]$	Not established
Latent selector combined vs. explicit	$+0.0261$	$[+0.0033, +0.0617]$	DeepSeek robust-positive; cross-judge unmet
Sham selector combined vs. explicit	$+0.0144$	$[-0.0050, +0.0500]$	Not established
Permuted selector combined vs. explicit	$+0.0139$	$[-0.0011, +0.0356]$	Not established
Latent combined vs. format-only	$+0.0372$	$[-0.0383, +0.1106]$	Not established

Table 8 preserves the broad analysis's contrast labels and estimates. Its `selector_latent_combined_vs_explicit` key and the targeted audit's `combined_vs_explicit` key both compare `latent__combined` with `latent__explicit` on the latent generation condition; "explicit" in these keys is the prompt-derived selector used in the reader-facing text. The two deliveries use the same 900 selected-string pairs but separate DeepSeek judging runs. The broad estimate is $23.5/900 = +0.0261$, with $[+0.0033, +0.0617]$; its order-consistent sensitivity is $+0.0250$, with $[+0.00335, +0.05889]$, over 892 resolved pairs. The targeted dual-order estimate is $21.5/900 = +0.0239$, with $[+0.0033, +0.0583]$. In both deliveries, the 872 exact-string ties contribute zero. Twelve of the 28 nonidentical pair deltas differ across the judging runs, and the unresolved counts are eight and 14, respectively. Thus $+0.0239$ is not the order-consistent reduction of $+0.0261$: the values are separate automated-judge sensitivities over identical model responses, not interchangeable summaries or independent experimental replications. For the corresponding broad-matrix Phi contrast, the estimate is $+0.0144$ with $[-0.0078, +0.0500]$ and is not established. The headline Phi interval $[-0.0067, +0.0500]$ belongs only to the targeted audit.

G.2 Cross-judge status

The canonical-versus-sham generation contrast is positive under DeepSeek ($+0.1794$, $[+0.0422, +0.3150]$) and Phi ($+0.2356$, $[+0.0933, +0.3850]$). Because most prompts have no accepted latent source, this agreement supports the canonical guidance condition relative to the generic sham, not a latent-specific scoring increment. The direct selector point estimates are positive under

both judges. Under the targeted audit's interval-and-Holm rule the result is positive under DeepSeek and not established under Phi; the broad analysis likewise does not establish the selector increment under Phi. Within the broad DeepSeek analysis, however, both the primary $+0.0261$ estimate and the $+0.0250$ order-consistent sensitivity are positive under the declared rule. Cross-judge support remains unmet.

G.3 CONTROLLED PANEL

Table 9: Selected controlled-panel estimates under the primary analysis.

Contrast	Mean delta	Two-way 95% interval	Status
Generation latent vs. explicit	$+0.1302$	$[-0.0938, +0.3906]$	Not established
Generation latent vs. sham	$+0.2031$	$[-0.0625, +0.5417]$	Not established
Generation latent vs. permuted	$+0.1406$	$[-0.1250, +0.4063]$	Not established
Combined selector vs. explicit	-0.0156	$[-0.1875, +0.1302]$	Not established
Raw combined selector vs. explicit	-0.0885	$[-0.3125, +0.0156]$	Not established
Combined vs. format-only	-0.0313	$[-0.2813, +0.1563]$	Not established

No declared controlled contrast satisfies the robust-positive criterion. The wide intervals are compatible with both improvements and regressions; these data establish neither equivalence nor a consistently adverse effect.

G.4 TEACHER-RATIONALE PANEL

The intention-to-treat analysis spans all 300 prompts; the prespecified eligible analysis contains 21. Neither population yields a robust positive generation or selector contrast after response-order and multiplicity checks. Several eligible-subgroup point estimates are adverse under both judges. The tested filtered-rationale intervention does not provide robust evidence of improvement under the declared analyses. The strings are filtered text, not executable teacher programs, so the finding concerns this implementation rather than ruling out a general rationale-prompting explanation.

Table 10: Teacher-rationale contrasts under the primary DeepSeek analysis. All rows fail the declared robust-positive criterion after response-order and Holm checks; the eligible subset was fixed by the rationale filter before outcomes. Trailing zeros preserve the source summaries' three-decimal precision.

Population	Contrast	Mean delta	Two-way 95% interval
ITT, 300	Rationale vs. canonical generation	-0.0360	$[-0.0880, +0.0100]$
ITT, 300	Rationale vs. explicit generation	-0.0290	$[-0.0710, +0.0080]$
ITT, 300	Total rationale-combined vs. explicit	-0.0370	$[-0.0870, +0.0070]$
Eligible, 21	Rationale vs. canonical generation	-0.3410	$[-0.7380, +0.0240]$
Eligible, 21	Rationale vs. explicit generation	-0.4370	$[-0.9760, +0.0710]$
Eligible, 21	Total rationale-combined vs. explicit	-0.5560	$[-1.1670, +0.0790]$

H MINIMAL SELECTOR MECHANISM

The strongest mechanism test asks whether adding accepted latent constraints changes the response selected from a fixed candidate pool. It does not compare against a separately generated explicit pool. Table 11 gives the complete decision census.

H.1 LENGTH AND PROFILE SENSITIVITY

Across the 341 combined-versus-exact disagreements, combined responses are 3.2375 words longer on average. Judge score correlates with response length at 0.5398 for DeepSeek and 0.6518 for Phi.

Table 11: Fixed-pool selector census.

Event	Count	Interpretation
Combined = prompt-derived	872	Adding accepted latent scores does not change the string
Combined $\neq$ prompt-derived	28	Minimal latent-difference set
Exact mode selects candidate 0	841	Baseline is largely a fixed-position selector
Combined = exact mode	559	Same selected string
Combined $\neq$ exact mode	341	Larger but weak-baseline diagnostic set

One decision-support profile accounts for 89.94% of signed DeepSeek improvement and 72.73% of signed Phi improvement. In the 15 equal-word-count pairs, both judges yield negative mean deltas. The observations are not a causal adjustment, but they identify plausible alternative explanations for the large exact-mode contrast.

H.2 What the direct selector comparison measures

The prompt-derived selector already includes constraints recoverable from the input. The combined selector differs only by accepted latent sources. Their disagreements therefore isolate the operational contribution of the latent layer more directly than canonical-versus-sham generation or combined-versus-exact selection. The selection-change rate is 3.11%, with 96.89% of pools retaining the prompt-derived choice. This quantifies selection reach, not the validity of the baseline or the benefit of each change. The 28 disagreements form the changed-response subset, while the aggregate outcome denominator remains 900 pools.

I Judge and response-order dependence

Table 12: Response-order census over the active pair corpus.

Quantity	Count
Pair records	48,912
Exactly equal pairs	35,262
Nonidentical pairs	13,650
Conflicting or unresolved nonidentical pairs	5,254
Conflict/unresolved rate among nonidentical pairs	38.49%

The rate in Table 12 combines outright direction flips with cases that remain unresolved after evaluating both orientations. It is not an estimate of annotator disagreement because the evaluators are models. Its role is narrower: it shows that a single response order is not a stable basis for the scientific claim.

Under the targeted audit, DeepSeek meets the interval-and-Holm rule ($+0.0239$, $[+0.0033, +0.0583]$, Holm-adjusted $p \approx 0.011$), while Phi does not ($+0.0144$, $[-0.0067, +0.0500]$). Under the broad matrix, DeepSeek is robust-positive within judge ($+0.0261$ primary; $+0.0250$ order-consistent), but Phi remains not established ($+0.0144$, $[-0.0078, +0.0500]$). Cross-judge support is unmet under both deliveries. This difference in threshold status is not a paired test of a difference between evaluator effects (**?**). The corpus-wide 38.49% rate is not a separately measured rate for the 28 direct-selector disagreements. Future evaluation should add blinded human judgments and task-grounded outcomes rather than simply adding more model judges to the same rubric.

J BENCHMARK, ROUTING, AND THRESHOLD EXTENSIONS

J.1 OBJECTIVE BENCHMARK CONFIGURATIONS

The objective results are mixed. The full GSM8K Qwen3-0.6B seed-13 run has a positive estimate whose interval excludes zero; the IFEval, IFBench, three-seed GSM8K subset, and cross-backbone subset intervals span zero. The full-set and hash-300 populations differ, so their contrast cannot isolate a seed effect. Table 13 retains all configurations and their original estimates; Table 4 promotes these measurements to the main text.

Table 13: Objective benchmark boundaries for blind-full selection versus the explicit-only baseline. IFEval and IFBench use official strict prompt accuracy. GSM8K uses normalized exact match.

Benchmark	Student / slice	n	Delta	95% interval	Status
IFEval	Qwen3-0.6B, full	541	$+0.0111$	$[-0.0296, +0.0518]$	Not established
IFBench	Qwen3-0.6B, full	300	-0.0100	$[-0.0333, +0.0100]$	Not established
GSM8K	Qwen3-0.6B, seed 13 full	1319	$+0.0273$	$[+0.0114, +0.0432]$	Positive in one seed
GSM8K	Qwen3-0.6B, three-seed hash-300	300	$+0.0111$	$[-0.0100, +0.0333]$	Inconclusive subset
GSM8K	Qwen2.5-1.5B, hash-300	300	-0.0233	$[-0.0700, +0.0233]$	Negative point; inconclusive
GSM8K	Llama-3.2-3B, hash-300	300	$+0.0133$	$[-0.0267, +0.0533]$	Not established

J.2 LEARNED GATE

A ridge model is trained on 841 completed IFBench and IFEval records and frozen before GSM8K outcomes are read. Its selected penalty is 100.0. Under the rule "select blind IDOL iff predicted paired gain is positive," the cross-validated selection rate is 0.9191 and the realized policy gain is $+0.0048$. Nearby penalties select IDOL in 0.9144 of cases and yield $+0.0059$. The model is therefore nearly always-on; it does not provide the selectivity needed to support a portable routing claim.

Table 14: Frozen ridge-gate sensitivity.

Ridge penalty	Selection rate	Realized gain
0.01	0.9144	$+0.0059$
0.1	0.9144	$+0.0059$
1	0.9144	$+0.0059$
10	0.9144	$+0.0059$
100	0.9191	$+0.0048$

J.3 THRESHOLD INTERPRETATION

LDTI combines latent coverage, distinction, candidate spread, and verifier signal. Sweeps do not yield one threshold that transfers across the fixed public slice, controlled cases, and natural benchmarks. A threshold chosen on one validation distribution must therefore be considered part of that distribution's analysis, not a reusable property of the method. Oracle or outcome-selected thresholds are sensitivity upper bounds and should not be reported as deployable policies.

K LATENCY AND RESOURCE ACCOUNTING

The denominator materially changes the interpretation. Relative to explicit best-of-16, cached IDOL adds about 15.86%; relative to a single generation, it more than doubles the stage sum. Neither ratio is a service-level latency: the measurements exclude synchronized request queuing, network transit, token streaming, batching effects, and end-to-end p50/p95 aggregation.

Table 15: Frozen four-A100 model-resident stage sums.

Configuration	Time (ms)	vs. explicit Bo16	vs. $N = 1$
$N = 1$ generation	3,767.58	–	1.0000×
Explicit best-of-16	7,023.78	1.0000×	1.8643×
Cached IDOL best-of-16	8,137.45	1.1586×	2.1599×
Cached IDOL with revision	8,534.86	1.2151×	2.2653×

L Reproducibility and Lineage

The anonymous repository for this submission is https://anonymous.4open.science/r/IDOL-ICLR27-8B0A.

L.1 Reproducibility checks

The artifact includes source code, frozen input slices, selected outputs, pairwise orientations, result summaries, model snapshot identifiers, manifests, and hashes. A one-command verifier recomputes current-response fingerprints and equality contracts; validates expected pair counts; rejects missing, unexpected, or duplicate records; checks privacy sanitation; and confirms the authoritative evidence boundaries.

Table 16: Lineage and reproducibility checks.

Check	Result
Stale orientation fingerprints	0
Pair/orientation equality failures	0
Missing pair records	0
Unexpected pair records	0
Duplicate pair records	0
Generated candidates traced	70,560
Selected variants traced	67,320
Reported provenance issues	0
Dependency-light tests	94 passing
Locked regenerated output files	14 exact
Regenerated numerical fields	10,489 exact

Table 17: Execution coverage and annotation status. "Complete" describes artifact coverage, not external validity.

Item	Count	Status
Generated candidates	70,560	Traced; no reported provenance issue
Selected variants	67,320	Traced to reported strings
Aggregate pair records	48,912	Complete response-string lineage
Response-order records	97,824	Complete; equality contracts pass
Human judgment rows	3,093	Blank; no human outcome evidence

L.2 Binding comparisons to response strings

Candidate pools may be regenerated after a detector or prompt repair. If pairwise judgments remain keyed only by historical indices, a result can refer to a response that is no longer selected. The final pipeline binds every pair orientation to fingerprints of the delivered strings, then recomputes equality from those strings. This is the appropriate unit of integrity for a selection study.

L.3 Scope of reproducibility

The reported automated analysis is reproducible under the recorded model snapshots and inputs. Reproducibility does not guarantee semantic validity, human relevance, or portability to a new model release. Third-party model weights are referenced rather than redistributed, and a fresh run requires access to those exact snapshots.

M Human-evaluation protocol and annotation status

The artifact defines blinded forms for pairwise content quality, constraint compliance, source-label relevance, and adjudication. It contains 18 forms and 3,093 rows, but every judgment cell is blank. These files demonstrate that a protocol can be instantiated; they do not constitute human validation.

A suitable follow-up should recruit annotators independently of the system developers, randomize response order, hide condition and selector identity, separate task quality from constraint compliance, and adjudicate disagreements. The minimal 28-pool latent-difference census is especially valuable because it tests the only decisions changed by latent additions. A larger population sample is also necessary: annotating only disagreement cases estimates the quality of changed decisions, not the overall deployment effect.

We recommend four reported quantities: preference for the combined response among the 28 disagreements; compliance difference under source-blind rubrics; inter-annotator agreement with uncertainty; and overall intent-to-treat effect after assigning zero change to the 872 ties. Until these exist, terms such as "human-aligned," "preferred," or "validated" are not supported.

N Models, data, and licenses

N.1 Models

The headline panels use the frozen snapshots in Table 18. Qwen model families are cited through their technical reports (**??**); the DeepSeek judge is described by **?**. Model-family names do not imply a positive result for every source–student pairing. The cross-backbone benchmark extension uses a Qwen2.5-1.5B model and a Llama-3.2-3B model; the latter is specified by its own model card (**?**), not by the earlier Llama 3 family report (**?**).

Model scope and unevaluated extensions. Model-family membership does not make checkpoints interchangeable. In particular, Phi-4 (**?**) and Phi-4-mini-instruct (**?**) are not the Phi-3.5-mini judge evaluated here, and Llama 3.3 (**?**) is not the Llama 3.2 benchmark backbone. DeepSeek-V3 (**?**) is likewise not the distilled DeepSeek-R1 judge. These separate releases provide context for replication design rather than evidence of additional model coverage.

Gemma 3 (**?**), Qwen3.5 (**?**), and Ministral 3 (**?**) within the Mistral 3 release (**?**) provide further settings for studying whether constraint compilation and fixed-pool attribution transfer across model families and modalities. They are not evaluated in this paper; no portability or multimodal result is inferred from their release reports.

Table 18: Exact model snapshots used by the headline pipeline.

Role	Repository identifier	Revision
Candidate generator	`Qwen/Qwen3-0.6B`	`c1899de289a04d12100db370d81485cdf75e47ca`
Semantic labeler/clusterer	`Qwen/Qwen3-32B-AWQ`	`0499c3ac83fdef8810b907a23894ba91e95eddd8`
Primary pairwise judge	`deepseek-ai/DeepSeek-R1-Distill-Qwen-7B`	`916b56a44061fd5cd7d6a8fb632557ed4f724f60`
Sensitivity judge	`microsoft/Phi-3.5-mini-instruct`	`2fe192450127e6a83f7441aef6e3ca586c338b77`

The fixed public panel samples eight candidates per prompt–condition–seed at temperature 0.8, top-$p = 0.95$, a 512-token cap, disabled thinking, and at most eight visible retries for scanner violations.

Both pairwise judges use temperature zero, top-$p = 1$, a 1024-token output cap (2048 for strict format recovery), and score both response orders. Their shared rubric evaluates directness, factual and internal soundness, usefulness and relevance, uncertainty handling, and compliance with requested format and scope; it explicitly forbids rewarding verbosity by itself.

N.2 Datasets

UltraChat supplies the base prompts for the fixed public slice (**?**); the slice is preselected and augmented with synthetic profiles. IFEval and IFBench provide verifiable instruction-following boundaries (**??**). GSM8K provides an arithmetic cross-domain boundary (**?**). Dataset licenses and redistribution conditions follow their respective releases; the artifact contains only the permitted slices and derived records.

Complementary datasets and evaluation targets. OpenAssistant's human-authored OASST1 conversations (**?**) are an alternative to the synthetic conversation source used here, not an additional IDOL dataset. MMLU (**?**), MATH (**?**), and TruthfulQA (**?**) would probe broad subject knowledge, competition mathematics, and the tendency to reproduce human misconceptions, respectively. Such outcomes could help separate behavioral compliance from task competence and truthfulness. PromptRobust studies adversarial prompt perturbations (**?**), while PromptBench provides a unified evaluation library (**?**). The nuance-oriented instruction-following reliability study of **?** provides a further prompt-variation perspective. None of these complementary datasets, robustness benchmarks, or toolkit evaluations is included in the reported IDOL outcome panels.

N.3 Privacy and anonymity

Released artifacts omit private paths, hostnames, usernames, account identifiers, raw execution logs, and other material that could compromise anonymous review. These removals do not change the prompts, generated responses, verifier outputs, statistical summaries, or reported outcomes.

O Source and learning extensions

O.1 Teacher-authored programs and distillation

A frozen teacher g can be an additional source h for the same program representation. The source contract is given in Appendix E; its executable output would require a separate end-to-end evaluation. For comparison, standard teacher–student distillation (**?**) uses an imitation objective such as

$$\mathcal{L}_{\mathrm{KD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\left[\ell(f_\theta(x), g(x))\right]. \tag{8}$$

Here the teacher output is part of the target. The present inference study instead evaluates canonical and prompt-derived constraints on student-generated responses. Source modularity permits future comparisons but does not imply that a teacher source improves on the canonical library, or that the library upper-bounds teacher performance.

O.2 Preference training

High-margin fixed-pool comparisons could supply student preference pairs for a separately validated objective, such as direct preference optimization (**?**). The current public semantic panel does not provide a validated pair set for a trained student, and no trained checkpoint is reported. Preference training therefore remains an interface rather than an empirical contribution of this study.

O.3 Verifier distillation

A compact IDOL-Ver could learn to approximate the deterministic and semantic checker stack. Such a verifier would need held-out calibration, cross-judge analysis, and human validity checks. Syntactic execution or agreement with a teacher judge is insufficient because the present results already show substantial evaluator and response-order dependence.

O.4 SELF-REVISION

Candidate revision can expose failed checks to the student and request a new response, related to iterative feedback-and-refinement approaches such as Self-Refine (**?**). The latency table includes a revision stage sum, but the evidence does not establish that revision improves task or human outcomes. A future study should compare revision against an equal-compute fresh sample and report both success and regression cases.

O.5 CONSTRAINED DECODING

Lexically constrained decoding can require words or phrases through grid beam search (**?**) or dynamic beam allocation (**?**). PICARD instead uses incremental parsing to reject inadmissible continuations for structured output, demonstrated on text-to-SQL (**?**). These are distinct constraint classes: lexical inclusion is not JSON-schema validation, and syntactic validity does not certify epistemic or scope requirements. Any IDOL-DECODE extension should specify the language or schema actually enforced, distinguish structural validity from open-ended answer quality, and measure latency under matched decoding budgets.

P THEORETICAL PERSPECTIVE

The framework can be viewed as scored selection over a finite candidate set. Let $U(x, y)$ denote unobserved task utility and $S_C(x, y)$ the combined checker score. A change in the maximizing score is only an operational intervention; its utility depends on the resulting response difference. Formally, for $y_C = \arg\max_y S_C(x, y)$ and an explicit baseline $y_E = \arg\max_y S_E(x, y)$, Equation 3 in the main text states this identity. The identity separates decision frequency from conditional benefit. The empirical first factor is $28/900$, or 3.11%, for the direct latent comparison. Conditional utility differences must be weighted by that frequency to obtain the full-panel effect; the identity alone does not assign a positive or negative utility to the changed responses. Moreover, automated judges are noisy proxies for U, so their conditional preference must be validated independently.

Suppose an automated judge J reverses a true pairwise preference with rate ϵ and exhibits orientation-dependent error ρ. Repeated model judging can reduce sampling variance but cannot identify systematic ϵ or ρ without an external outcome. The observed 38.49% conflict/unresolved rate among nonidentical pairs is thus a warning about measurement stability, not an estimate of true error.

Finally, routing can help only if a score predicts the sign of $U(x, y_C) - U(x, y_E)$ out of sample. The observed 91.91% selection rate is close to always-on, while the reported realized gain is $+0.0048$. The rate alone does not identify which regressions are prevented. A prospective routing test must evaluate gains and regressions on held-out data against both always-on and explicit-only policies.

Q EXTENDED POSITIONING

Answer-free versus rationale transfer. Rationale distillation transfers an explanation or reasoning trace that is coupled to a target response. IDOL is designed to transfer a behavioral program while letting the student produce all response content. The present study tests canonical and prompt-derived programs rather than the full teacher-authored path. The distinction is meaningful only when leakage controls work and the program contributes beyond constraints recoverable from the prompt. The 28-pool direct selector test is designed to measure precisely this increment.

Reranking versus latent discipline. Reranking is the operational mechanism used in the current evidence, but it is not by itself the novelty. Our exact-normalized full-response mode diagnostic is also not answer-level self-consistency, which aggregates answers across sampled reasoning paths (**?**). Any score can rerank candidates. The scientific question is whether an answer-free latent source adds valid signal beyond prompt-derived constraints. The current evidence quantifies a 3.11% choice-change rate and positive aggregate point estimates under both judges, with the cross-judge support criterion unmet. This separates operational reach from outcome corroboration.

Selective prediction versus selective constraint use. Classical selective prediction decides whether to emit a prediction. LDTI instead summarizes prerequisites for applying an additional latent scoring layer; useful routing also requires prediction of incremental benefit. The fallback still answers using explicit constraints. This makes the routing target more subtle: it must predict the incremental value of one scorer over another rather than confidence in a single output.

Constraint programming versus model alignment. Constraint programming offers inspectability because requirements can be listed and checked. Alignment claims require more: the checks must correspond to outcomes people or tasks value. The human-evaluation templates define a possible validation protocol, but no judgments were collected. Automated evaluation therefore supports the reported measurements rather than a human-alignment conclusion.

R BROADER IMPACTS AND SAFEGUARDS

Answer-free behavioral programs could reduce the need to expose proprietary teacher answers and may help small models preserve schemas, abstain under missing evidence, or respect scope. These benefits would be useful for on-device and privacy-sensitive settings if independently validated.

The same mechanism can also encode a teacher's biases in refusal, certainty, or scope. A semantic verifier may reward verbosity, a synthetic profile may favor one communication style, and a weak router may apply the latent layer almost everywhere. These risks are especially important when a small model is used in high-stakes domains.

Practical safeguards include source labels, deterministic checks where possible, explicit fallback behavior, dual-order judge evaluation, retention of adverse outcomes, and human validation before deployment. Sensitive uses should also measure subgroup effects, failure costs, and service-level latency under the actual deployment stack.

S REPRODUCIBILITY CHECKLIST

Table 19: Compact checklist for reproducing the reported evidence.

Step	Required check
Inputs	Verify fixed-slice, condition, seed, and model-snapshot hashes
Generation	Confirm candidate count, prompt condition, and scanner receipts
Selection	Recompute prompt-derived (S_E) and combined (S_C) scores
Lineage	Bind selected strings to pair fingerprints and recompute equality
Judging	Evaluate DeepSeek and Phi in both response orders
Statistics	Regenerate paired intervals, Holm families, and all 14 outputs
Boundaries	Retain controlled, rationale, benchmark, routing, and latency results
Human evidence	Confirm that blank forms are not counted as outcomes
Reporting	Separate execution integrity from semantic or human validity

The key interpretive check is last: an exact rerun of an automated pipeline reproduces its measurements, not the truth of the construct it measures.