✦ SLMJURY: Can Small Language Models Judge as Well as Large Ones?

Anonymous EMNLP submission

Abstract

Large language models (LLMs) are widely used as judges for evaluating model outputs, but their high cost, latency, and opacity limit scalability. We introduce SLMJURY, a framework for evaluating small language models (SLMs) as judges across two paradigms: **closed-ended** binary correctness and **open-ended** quality scoring. We benchmark **16 SLM judges** (0.6B–14B parameters) from four model families across **ten benchmarks**: eight closed-ended tasks spanning mathematical, scientific, and general reasoning (N=64,824 judgments per configuration), plus SummEval and MT-Bench for summarization and conversational scoring. We formalize judging as a budget-conditioned function and study five dimensions. Four findings emerge. **(1)** The overthinking effect is *domain-dependent*: for most judges quick 10-token verdicts match or beat extended reasoning on mathematical judging (by 2–7% where they help), while reasoning wins on general tasks by up to 23%. **(2)** Domain generalization separates model families, with math-to-general accuracy gaps ranging from under 10% to nearly 40%. **(3)** Closed-ended and open-ended judging draw on different capabilities: the best binary judge (Phi-4) drops to rank 9 on MT-Bench, while reasoning-trained models invert this ordering. **(4)** Under the Reflect-Critique-Refine (RCR) debate protocol, multi-agent debate degrades accuracy across all tested configurations, whereas the top judges resist six adversarial personas with $\leq$0.55% variance. Reliable automated evaluation does not require large proprietary models, yet no single SLM dominates.

1 Introduction

Large language models (LLMs) are now the default automated judges for evaluating model outputs (Zheng et al., 2023; Kim et al., 2024; Zhu et al., 2025). Systems such as MT-Bench (Zheng et al., 2023) and JudgeLM (Zhu et al., 2025) rely on proprietary models like GPT-4 (Achiam et al., 2023) to score student solutions, achieving high agreement with human annotations. However, consider the cost structure of this paradigm. For a proprietary judge processing N solutions at token budget B, the total evaluation cost scales as $C_{\text{total}} = N \cdot (\alpha B_{\text{in}} + \beta B)$, where α, β are per-token input/output prices and B_{in} is the prompt length. At evaluation scale ($N > 10^4$), this cost becomes a bottleneck for workflows like reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) and constitutional AI (Bai et al., 2022). Beyond cost, the proprietary paradigm introduces privacy risks (inputs leave the local network), result opacity (model versions change silently), and latency (Wang et al., 2024b; Chen et al., 2024; Szymanski et al., 2025).

Recent small language models (SLMs) provide a way to reduce both C_{eval} and B simultaneously. Models such as Phi-4 (Abdin et al., 2024), Phi-4-mini (Abouelenin et al., 2025), Phi-4-Reasoning (Abdin et al., 2025), Qwen 2.5 (Qwen et al., 2025), Qwen 3 (Yang et al., 2025), and LLaMA 3 (Grattafiori et al., 2024) achieve strong reasoning at a fraction of the cost, running locally on a single GPU. Prior work shows that SLMs can match LLM-level performance on reasoning (Srivastava et al., 2025b), benchmark-free evaluation (Srivastava et al., 2025c), and agentic generation (Srivastava et al., 2026). This raises a natural question: *if SLMs can reason, can they also judge?* Despite growing interest in LLM-as-a-judge (Bi et al., 2026; Jayarao et al., 2025; Li et al., 2026), no study has systematically evaluated SLMs as judges across diverse task domains and evaluation paradigms, spanning the full space of token budgets, persona conditions, debate protocols, and both closed-ended and open-ended evaluation.

We address this gap with SLMJURY, a framework for evaluating SLMs as judges across two complementary paradigms. For **closed-ended** eval-

Table 1: **Feature comparison of SLMJURY with related LLM-as-a-Judge work.** To our knowledge, SLMJURY is the first framework to evaluate off-the-shelf SLM judges across all seven evaluation dimensions. See Appendix N for dataset details and Appendix O for model configurations.

Feature	MT-Bench	JudgeLM	Prometheus 2	JudgeBoard	JudgeBench	PoLL	Explicit Reason.	SLMJURY
Off-the-shelf SLM judges ($\leq$14B)†	✗	✗	✗	✓	✗	✗	✓	✓
Binary correctness eval.	✗	✗	✗	✓	✓	✗	✗	✓
Open-ended quality scoring	✓	✗	✓	✗	✗	✓	✗	✓
Token budget analysis	✗	✗	✗	✗	✗	✗	✓	✓
Persona robustness	✗	✗	✗	✗	✗	✗	✗	✓
Ensemble / majority voting	✗	✗	✗	✗	✗	✓	✗	✓
Multi-agent debate	✗	✗	✗	✓	✗	✗	✗	✓
Multi-family evaluation	✗	✗	✗	✓	✓	✓	✗	✓
Num. judge models	1	3	2	8	14	3	3	**16**
Num. benchmarks	1	2	8	5	1	6	1	**10**

† Non-fine-tuned models with $<$14B parameters. JudgeLM and Prometheus 2 train *custom* judges (7–33B and 7B/8$\times$7B); PoLL uses off-the-shelf models but includes proprietary LLMs (GPT-3.5, Claude-3-Haiku) rather than SLMs.

uation, we define a judge as a function $f_{\mathcal{J}}^{(B)} : \mathcal{Q} \times \mathcal{G} \times \mathcal{R} \rightarrow \mathcal{V}$ that maps a question q, ground truth reference g, and student reasoning r_s to a verdict $v \in \{\texttt{Correct}, \texttt{Incorrect}\}$ under token budget B. For **open-ended** evaluation, we extend to quality scoring, where judges assign numerical scores compared against human annotations to measure SLM-Human correlation (SummEval) or against large oracle model scores to measure SLM-LLM correlation (MT-Bench). We constrain $\mathcal{J}$ to models with $|\theta_{\mathcal{J}}| \leq 14$B parameters, evaluate under two budget settings ($B \in \{10, 8192\}$), and study five complementary evaluation dimensions (Figure 1). Our contributions:

(1) We benchmark **16 SLM judges** from four families spanning 0.6B–14B parameters across **ten benchmarks** (8 closed-ended + SummEval + MT-Bench, N=64,824 judgments per configuration, $>$3,900 experiments), with the best judge reaching 89.55% on closed-ended evaluation (§4.2). **(2)** We show the overthinking effect is *domain-dependent*: quick verdicts (B=10) outperform reasoning on math judging for 8 of 13 dual-setting judges, while reasoning outperforms on general tasks for 8 of 13, with single-dataset swings up to 23% (§4.2). **(3)** We conduct what is, to our knowledge, the first cross-paradigm analysis of the same SLM judges: the best binary judge (Phi-4) drops to rank 9 on MT-Bench (ρ=0.21), while Phi-4-Reasoning rises to rank 1 (ρ=0.57), indicating that closed-ended and open-ended judging draw on different capabilities (§4.5). **(4)** The top judges resist adversarial persona manipulation ($\leq$0.55% variance), while under the RCR debate protocol, debate *degrades* accuracy across all tested configurations (§4.3, §4.4).

Across all experiments, our results show that reliable automated evaluation does not require large proprietary models, but that no single SLM excels across all evaluation paradigms.

2 Related Work

LLM- and SLM-as-a-Judge. MT-Bench and Chatbot Arena (Zheng et al., 2023) established GPT-4 as a reliable proxy for human evaluation, and JudgeLM (Zhu et al., 2025), Prometheus (Kim et al., 2024), and PandaLM (Wang et al., 2024b) explored open-source alternatives, while SummEval (Fabbri et al., 2021) and MT-Bench supply human-annotated and oracle-scored benchmarks for open-ended scoring. Yet LLM judges degrade on expert tasks (Szymanski et al., 2025) and exhibit position and verbosity biases (Chen et al., 2024; Gu et al., 2024; Li et al., 2025; Yang et al., 2026). This has driven a turn toward smaller judges: JudgeBoard (Bi et al., 2026) benchmarks SLMs for reasoning, JudgeBench (Tan et al., 2025) shows even GPT-4o is near-random on hard pairs, and Wang et al. (2026) show small reward models can outperform LLM-as-a-judge baselines via backward inference, with further work on representation-based judging (Li et al., 2026), code (Crupi et al., 2026; Jiang et al., 2025), contextual settings (Xu et al., 2025a), and training-based improvements (Xu et al., 2025b; Zhang et al., 2025b). Closest to us, Jayarao et al. (2025) study reasoning effects on Qwen 3 judges with RewardBench. *We complement these by showing the token-budget effect is domain-dependent across 16 judges and ten benchmarks spanning both paradigms.*

Token Budget, Overthinking, and Prompt Sensitivity. Longer reasoning does not always help: Han et al. (2025) introduce token-budget-aware prompting, Wang et al. (2024a) show strategy efficiency depends on task type, and Srivastava et al. (2025d) show extended chain-of-thought can hurt smaller models on math, with further work

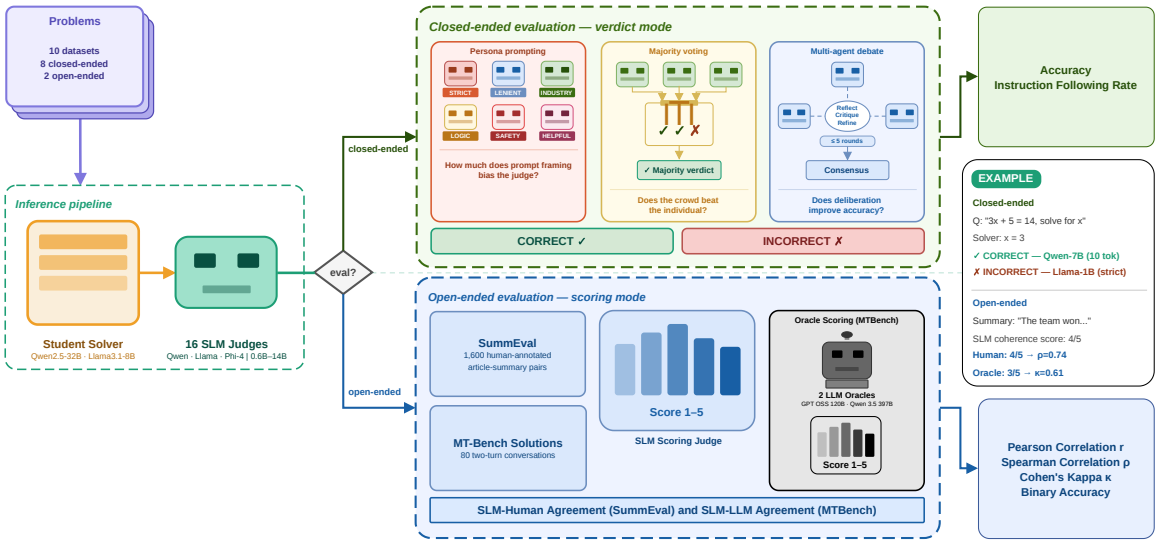

Figure 1: **Overview of the SLMJURY framework.** SLMJURY evaluates SLMs as judges across two paradigms: **closed-ended** binary correctness classification (8 benchmarks spanning math, science, and general reasoning) and **open-ended** quality scoring (SummEval and MT-Bench). It studies five dimensions: **(a)** individual judging at two token budgets, **(b)** persona-guided evaluation, **(c)** majority voting ensembles, **(d)** multi-agent debate, and **(e)** open-ended scoring. Evaluation spans 16 judges, 2 students, and 10 benchmarks.

reporting diminishing returns from test-time scaling (Sui et al., 2025; Ma et al., 2025; Ghosal et al., 2026; Shojaee et al., 2026; Aggarwal et al., 2025). A parallel line shows judges are sensitive to framing: persona prompts shift judge behavior (Tseng et al., 2024; Salewski et al., 2023; Abdullahi et al., 2026; Ryan et al., 2025). *We bring both lenses to judging and find the overthinking effect is domain-dependent, while the top SLM judges resist six adversarial personas with ≤1% variance.*

Ensembles and Multi-Agent Debate. Self-consistency (Wang et al., 2022) and repeated sampling (Brown et al., 2024) improve reasoning, SLM ensembles can match larger models (Cho et al., 2025), and PoLL (Verga et al., 2024) shows diverse small panels can outperform a single large judge; debate improves factual accuracy and divergent thinking (Du et al., 2024; Liang et al., 2024), with ChatEval (Chan et al., 2024) showing multi-agent debate improves open-ended evaluation and Srivastava et al. (2025a) introducing RCR (Hu et al., 2026; Lin et al., 2025; Alam et al., 2026). *We find ensembles add only +0.06% once accuracy saturates near 89%, and debate consistently degrades binary judging, echoing Wynn et al. (2025) and Zhang et al. (2025a).* Table 1 positions SLMJURY against this prior work: no existing study combines all seven dimensions in one framework, nor evaluates one set of SLM judges across both paradigms.

3 The SLMJURY Framework

SLMJURY is a modular framework for studying SLMs as judges across closed-ended and open-ended paradigms (Figure 1). We fix notation: $\mathcal{Q}$, $\mathcal{G}$, $\mathcal{R}$ denote the spaces of questions, ground-truth references, and student responses, and $\mathcal{V} = \{\texttt{Correct}, \texttt{Incorrect}, \texttt{Undefined}\}$ the verdict space. A judge $\mathcal{J}$ has $|\theta_{\mathcal{J}}| \leq 14$B parameters under a token budget $B \in \{10, 8192\}$. We describe the pipeline, judge inference, metrics, evaluation approaches, and open-ended scoring below; the full formal treatment is in Appendix E.

3.1 Evaluation Pipeline

The pipeline has three stages: student solution generation, judge evaluation, and metric computation. A *student model* $\mathcal{S}$ generates solutions to problems from benchmark $\mathcal{D}$. All student inference uses temperature τ_s=0.7, top-p=0.9, and B_s=1,024 tokens, with dataset-specific prompt templates (Appendix L). Inference runs via vLLM (Kwon et al., 2023) with PagedAttention (Yi et al., 2024). Each solution is stored in a structured schema preserving the problem, ground truth, and student response; details and the full schema appear in Appendix J.

3.2 Judge Inference and Token Budget

Following the standard reference-based verification paradigm used in MT-Bench (Zheng et al., 2023),

JudgeLM (Zhu et al., 2025), and PoLL (Verga et al., 2024), each judge $\mathcal{J}$ maps a triple $(q, g, r_s) \in \mathcal{Q} \times \mathcal{G} \times \mathcal{R}$, the question, the ground-truth reference, and the student's reasoning chain, to a verdict $v \in \mathcal{V} = \{\texttt{Correct}, \texttt{Incorrect}, \texttt{Undefined}\}$, with the reference set to $g = r_s^*$ when gold reasoning exists[1] and $g = a^*$ otherwise. We define the budget-conditioned judge function as composition

$$f_{\mathcal{J}}^{(B)} = \textsc{Parse} \circ \mathcal{D}_B \circ \mathcal{T}_B \;:\; \mathcal{Q} \times \mathcal{G} \times \mathcal{R} \to \mathcal{V}, \quad (1)$$

where $\mathcal{T}_B$ formats the budget-appropriate prompt, $\mathcal{D}_B$ decodes a response $y \sim \pi_{\mathcal{J}}(\cdot \mid \mathcal{T}_B(q, g, r_s))$ of at most B tokens, and $\textsc{Parse} : y \mapsto \mathcal{V}$ extracts the verdict via the six-level priority cascade of Appendix G. We instantiate $\mathcal{D}_B$ under two settings:

Quick verdict (B=10). The judge responds with a single word using deterministic decoding (τ=0). The prompt instructs the judge to focus only on the final answer. Verdict extraction applies a six-level priority cascade (Appendix G).

Reasoned verdict (B=8,192). The judge reasons step-by-step before emitting `\boxed{CORRECT}` or `\boxed{INCORRECT}`. Non-thinking models use deterministic decoding (τ=0); the same cascade extracts the verdict.

Thinking mode. For models with native chain-of-thought (Qwen3, Phi-4-Reasoning variants), thinking mode is enabled at B=8,192 with τ=0.6, top-p=0.95, top-k=20, min-p=0. Always-thinking models require sampling for their internal chain-of-thought and cannot produce valid outputs under deterministic decoding; at B=10 the entire budget is consumed by reasoning tokens before any verdict is emitted. They are therefore evaluated at B=8,192 only and excluded from Δ comparisons (Appendix O).

3.3 Ground Truth and Metrics

Judge verdicts are evaluated against a programmatic oracle: a verdict is `Correct` if $\textsc{Equiv}(\hat{a}, a^*, \delta) = \top$ and `Incorrect` otherwise, where $\hat{a}$ is the student's extracted answer (Appendix F) and $\textsc{Equiv}$ is a dataset-type-specific equivalence function (normalized numeric for GSM8K/GSM-Plus, symbolic for MATH, label matching for multiple-choice; full details in Appendix G). We report two primary metrics. **Accuracy** measures agreement with the oracle:

$$\mathrm{Acc}(\mathcal{J}, B) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\left[f_{\mathcal{J}}^{(B)}(q_i, g_i, r_{s,i}) = v_i^* \right] \quad (2)$$

Instruction Following Rate (IFR) measures the fraction of parseable (non-`Undefined`) responses (Appendix E gives the full formal treatment). All metrics are computed over N=64,824 judgments (2 students $\times$ 8 datasets) per configuration.[2]

3.4 Evaluation Approaches

Individual judging. All 16 judges are evaluated independently under both token budgets, yielding up to 32 configurations. Always-thinking models contribute one configuration each.

Persona-guided evaluation. Six adversarial system prompts are injected into top judges: *Strict*, *Lenient*, *Industry*, *Logic*, *Safety*, and *Helpfulness* (Appendix M). Other parameters are held constant.

Majority voting ensemble. We select the top-5 judges by best accuracy, using each model's best token setting, and enumerate all $\binom{5}{3}$=10 three-model juries. The ensemble verdict is the majority label among three judges, with ties defaulting to `Undefined` (Appendix B).

Multi-agent debate. We adopt the RCR (Reflect-Critique-Refine) protocol of Srivastava et al. (2025a) for binary correctness judging. Three agents produce independent verdicts (Round 0), then iteratively critique and refine over up to R=5 rounds. Consensus requires unanimity; otherwise the final verdict falls back to majority vote. We evaluate *Variant A* (cross-architecture, τ=0.0) and *Variant B* (same model, $\tau \in \{0.0, 0.4, 0.9\}$). The full algorithm and debate prompts are in Appendix I.

3.5 Open-Ended Evaluation

Beyond binary correctness, each judge also realizes a scoring map $s_{\mathcal{J}} : \mathcal{X} \to \{1, \ldots, 5\}$ that rates an item x (a summary or a conversation) on a five-point scale. We assess agreement between a judge's score vector $\mathbf{s} = (s_{\mathcal{J}}(x_i))_i$ and a reference vector $\mathbf{g}$, where $\mathbf{g}$ is human (SummEval) or oracle (MT-Bench); the headline objective is the Spearman rank correlation $\rho(\mathbf{s}, \mathbf{g})$, with Pearson r, Cohen's κ, accuracy, and mean squared error (MSE)

[1] Gold step-by-step reasoning ships with the datasets (GSM8K, GSM-Plus, MATH); we do not synthesize it.

[2] At N=64,824, the standard error near 89% is $\approx$0.12%; all reported differences exceeding $\pm$0.5% are significant at $p < 0.001$ by a two-proportion z-test.

as secondary measures (Appendix H). We use two benchmarks targeting complementary regimes:

SummEval (Fabbri et al., 2021) (SLM-Human Correlation). Judges score 1,600 machine-generated article-summary pairs on four dimensions (coherence, consistency, fluency, relevance) from 1 to 5, compared against human annotations to measure SLM-Human agreement (Appendix L.4).

MT-Bench (Zheng et al., 2023) (SLM-LLM Correlation). Judges score 80 multi-turn conversations on overall quality (1–5) across eight categories. Two large oracle models, GPT-OSS-120B (120B) and Qwen3.5-397B-A17B (397B MoE, 17B active), serve as reference scorers via the Together API, each scoring responses from both student models, yielding four oracle-student combinations per judge. Oracle and SLM judges use the *identical* scoring prompt, so correlation isolates capability, not prompt design (Appendix L.4).

Open-ended metrics. For SummEval we correlate SLM and human scores; for MT-Bench, SLM and oracle scores. Both use Pearson r (linear agreement), Spearman ρ (ordinal agreement), Cohen's κ (binary agreement after thresholding at score ≥ 4), classification accuracy, and mean squared error. Formal definitions and the score-parsing cascade are in Appendix H.

4 Results and Insights

We report findings across the five evaluation dimensions of Figure 1, with $\mathcal{J}$ ranging over 16 judges and accuracy measured against the oracle of §3.3.

4.1 Evaluation Setup

We evaluate 16 judges across 2 student models and 8 closed-ended datasets spanning three domains: mathematical reasoning (GSM8K, GSM-Plus, MATH), scientific reasoning (ARC-Easy, ARC-Challenge), and general reasoning (HellaSwag, WinoGrande, TruthfulQA), yielding N=64,824 judgments per configuration. The two student models are Qwen2.5-32B-GPTQ-Int8 (95.45% solver accuracy on GSM8K) and LLaMA-3.1-8B-Instruct (82.49%), providing a strong and a weaker solution source (Appendix J). Results are averaged over both student models and all datasets unless stated otherwise. Per-dataset breakdowns are in Appendix A. Hardware and implementation details are in Appendix K.

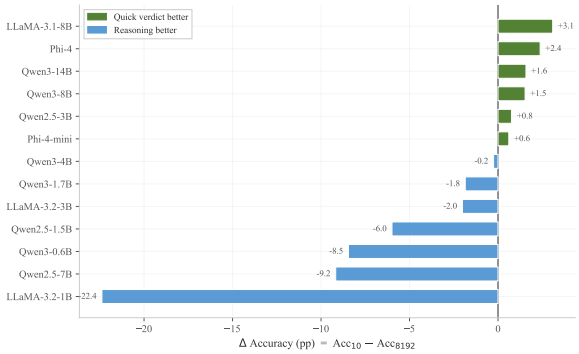

Figure 2: **Overthinking delta (Δ) across 13 dual-setting judges.** Green = quick verdicts (B=10) win; blue = reasoning wins. The full grouped-bar comparison appears in Appendix A.

4.2 Individual Judging and Token Budget Effects

SMALL MODELS ARE CAPABLE JUDGES ACROSS DOMAINS. **A 14B open judge (Phi-4, 89.55% at B=10) sustains near-90% agreement with the oracle across eight benchmarks spanning three domains, without any judge-specific fine-tuning.** As Table 2 and Figure 2 show across all 16 judges, Qwen3-14B follows at 89.51% and the best sub-5B model, Phi-4-mini, reaches 88.22%; even at 4B parameters, Qwen3-4B achieves 87.81% (B=8,192), trailing the best 14B model by only 1.74%. The decisive factor is not scale but *domain coverage*: top models score 92–97% on math yet only 76–80% on HellaSwag, the hardest general benchmark, which demands commonsense rather than verifiable computation. This single gap, not parameter count, explains most of the spread between judges (Figure 5, Appendix A).

THE OVERTHINKING EFFECT IS DOMAIN-DEPENDENT. **Quick verdicts beat extended reasoning on mathematical judging for 8 of 13 dual-setting judges, yet reasoning wins on general tasks for 8 of 13, so the two effects partially cancel in any single-number summary.** Let $\Delta(\mathcal{J}, \mathcal{D}) = \mathrm{Acc}(\mathcal{J}, 10) - \mathrm{Acc}(\mathcal{J}, 8192)$ on domain $\mathcal{D}$; the aggregate Δ masks two opposing regimes. On *math* (GSM8K, GSM-Plus, MATH), quick verdicts win ($\Delta_{\text{math}} > 0$) for 8 of 13 judges, with per-dataset gains of 2–7% for the judges that benefit, consistent with the view that correctness here is structurally verifiable (match a number or a symbol), so extra deliberation has limited upside. On *general* tasks (HellaSwag, WinoGrande, TruthfulQA) the sign flips for 8 of 13 judges ($\Delta_{\text{general}} < 0$): Qwen3-4B loses 11.63% on

Table 2: **Individual judge performance.** Accuracy and IFR (%) averaged over 2 student models and 8 datasets (N=64,824 per configuration). **Bold** = column best. † = always-thinking models (8,192 tokens only). Δ = Acc(B=10) − Acc(B=8192); positive ⇒ quick verdict wins.

			Accuracy (%)			IFR (%)	
Family	**Judge Model**	**Parameters**	**10 tokens**	**8192 tokens**	**Δ (%)**	**10 tokens**	**8192 tokens**
LLaMA 3.x	LLaMA-3.2-1B-Instruct	1B	43.15	65.53	−22.38	94.02	97.41
	LLaMA-3.2-3B-Instruct	3B	79.76	81.77	−2.01	100.00	99.92
	LLaMA-3.1-8B-Instruct	8B	86.79	83.70	+3.09	99.98	99.79
Qwen 2.5	Qwen2.5-1.5B-Instruct	1.5B	73.92	79.91	−5.99	100.00	99.21
	Qwen2.5-3B-Instruct	3B	77.65	76.89	+0.76	99.99	99.80
	Qwen2.5-7B-Instruct	7B	77.45	86.62	−9.17	99.99	99.94
Qwen 3	Qwen3-0.6B	0.6B	67.92	76.37	−8.45	98.06	95.31
	Qwen3-1.7B	1.7B	84.11	85.96	−1.85	99.98	99.99
	Qwen3-4B	4B	87.56	87.81	−0.25	**100.00**	**100.00**
	Qwen3-8B	8B	88.96	87.43	+1.53	**100.00**	**100.00**
	Qwen3-14B	14B	89.51	87.93	+1.58	**100.00**	**100.00**
Phi-4	Phi-4	14B	**89.55**	87.17	+2.38	99.98	**100.00**
	Phi-4-Reasoning†	14B	—	88.24	—	—	**100.00**
	Phi-4-Reasoning-Plus†	14B	—	**88.75**	—	—	**100.00**
	Phi-4-mini-Instruct	3.8B	88.22	87.61	+0.61	99.95	99.84
	Phi-4-mini-Reasoning†	3.8B	—	83.32	—	—	98.31

its general-domain average and 23.12% on WinoGrande alone (71.43 → 94.55), where pronoun resolution and commonsense inference are harder to settle in ten tokens. The sign of Δ flips *within the same judge* across domains, so it is driven by *task structure rather than model capability*: a single threshold separating "fast" from "slow" judges does not survive once benchmarks span verifiable and non-verifiable domains. The per-judge split is a majority (8 of 13) rather than unanimous, and the effect is clearest in the domain-averaged Δ and the per-dataset breakdown of Appendix A.

DOMAIN GENERALIZATION SEPARATES MODEL FAMILIES. **The math-to-general accuracy gap spans from under 10% to nearly 40% across families, so this gap, more than raw accuracy, separates a broadly applicable judge from a domain specialist.** Qwen 2.5 sits at the specialist end: Qwen2.5-7B reaches 95.9% on math but 58.7% on general reasoning at B=10, a 37.2% gap that we hypothesize reflects math-heavy instruction tuning, though architectural and data factors we did not control may also contribute. Phi-4 and Qwen 3 instead hold this gap near 10% (9.2–10.6%), so they remain comparatively reliable on benchmarks they were not specifically optimized for (Figure 3).

INSTRUCTION FOLLOWING IS NEAR-PERFECT. The instruction following rate (IFR) exceeds 99.8% for every top judge at both budgets, so the accuracy differences above reflect

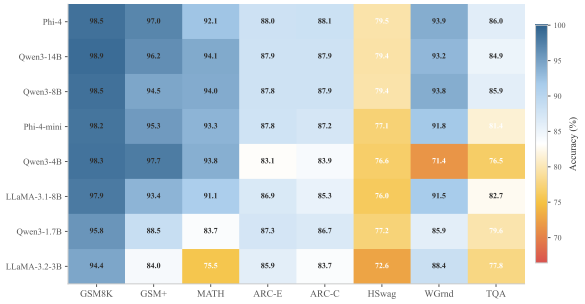

Figure 3: **Per-dataset judge accuracy at** B=10 for the top-8 judges. Math benchmarks cluster above 85%, while general-reasoning tasks expose cross-family gaps: Qwen 2.5 exceeds 90% on math but falls below 65% on general, whereas Phi-4 and Qwen 3 stay uniform (§4.2).

genuine disagreement with the oracle rather than unparseable output (Appendix A).

4.3 Persona Robustness

WELL-TRAINED JUDGES RESIST PERSONA MANIPULATION. **Qwen3-14B at B=10 moves by only 0.08% across all six adversarial personas (64,824 judgments each), so the personas we test do not redirect a strong judge's verdict.** Measuring robustness by the range $r(\mathcal{J}) = \max_p \mathrm{Acc}(\mathcal{J}, p) - \min_p \mathrm{Acc}(\mathcal{J}, p)$ over the six personas p, Table 3 and Figure 4 show that the top judges cluster tightly across all eight benchmarks: Qwen3-14B at r=0.08% (89.32 to 89.40), Phi-4 at r=0.55%, and Qwen3-4B at r=1.39%. Throughout, we report persona deltas against each judge's no-persona individual accuracy (the shaded

Table 3: **Persona effect on judge accuracy (%).** Shaded = base (no persona). Bold = best per row; underline = worst. Range = max − min across six personas. Full persona prompts are in Appendix M.

Judge	Tokens	Base	Strict	Lenient	Industry	Logic	Safety	Helpful	Range
LLaMA-3.1-8B-Instruct	10	86.79	86.67	76.59	86.29	**87.08**	85.44	86.17	10.49
LLaMA-3.1-8B-Instruct	8192	83.70	78.34	81.26	**84.49**	82.52	81.61	82.03	6.15
Qwen3-4B	10	87.56	89.02	89.71	88.35	**89.74**	89.43	88.93	1.39
Qwen3-4B	8192	87.81	87.71	87.79	**88.01**	87.91	87.89	87.84	0.30
Qwen3-14B	10	89.51	89.38	**89.40**	89.33	89.36	89.32	89.37	0.08
Qwen3-14B	8192	87.93	87.75	88.01	88.12	87.83	87.80	**88.24**	0.49
Phi-4	10	89.55	89.80	89.33	89.60	**89.88**	89.76	89.62	0.55
Phi-4	8192	87.17	**87.26**	85.90	87.11	86.03	87.12	87.06	1.36
Phi-4-mini-Instruct	10	88.22	86.86	86.93	86.82	87.51	**87.75**	86.69	1.06
Phi-4-mini-Instruct	8192	87.61	75.70	86.51	85.09	86.47	83.62	**87.44**	11.74

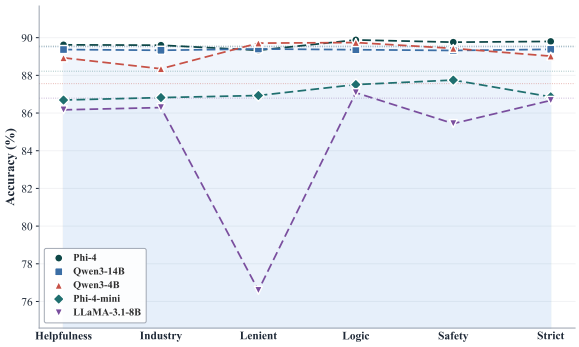

Figure 4: **Persona sensitivity at 10 tokens.** Dashed lines mark no-persona baselines. LLaMA-3.1-8B collapses under the Lenient persona (−10.20%); Qwen3-14B and Phi-4 remain nearly flat.

Base column of Table 3). This is consistent with these judges having internalized a stable correctness boundary rather than an instruction-following heuristic that surface framing can flip.

LENIENT AND STRICT ARE THE PRIMARY ATTACK SURFACES. **The Lenient persona collapses LLaMA-3.1-8B by 10.20%** (86.79 → 76.59) **but moves Qwen3-14B by only 0.11%, nearly two orders of magnitude smaller.** The Lenient prompt asks the judge to award partial credit and forgive errors; a weak judge appears to reinterpret this as "is the attempt reasonable?" and accepts wrong answers, while a strong judge is largely unaffected. The mirror-image failure appears under reasoning: at B=8,192 the Strict persona collapses Phi-4-mini by 11.91% (87.61 → 75.70), the largest single persona effect in the study, by pushing it to second-guess answers that are in fact correct.

EXTENDED REASONING CAN AMPLIFY PERSONA SENSITIVITY. **Moving from B=10 to B=8,192 widens Phi-4's persona range from 0.55% to 1.36%** (2.5×) **and Phi-4-mini's from 1.06% to 11.74%** (11.1×). A plausible reading is that each extra reasoning token is also extra surface for the persona to steer the chain of thought, so the deliberation that helps on general tasks (§4.2) can make a weaker judge more fragile under adversarial framing. The effect is most pronounced for the smaller Phi-4-mini.

Table 4: **Open-ended evaluation: top-5 judges by SummEval (SLM-Human) Spearman** ρ, with MT-Bench (SLM-LLM) ρ and closed-ended (CE) rank.

Judge	CE Rank	SE ρ	MB ρ	MB Rank
Phi-4	1	**0.520**	0.215	9
Phi-4-Reasoning-Plus	4	0.489	0.500	3
Qwen3-14B	2	0.487	**0.550**	2
Phi-4-Reasoning	5	0.479	0.570	**1**
Qwen3-8B	3	0.453	0.466	4

4.4 Majority Voting and Multi-Agent Debate

ENSEMBLES PROVIDE MARGINAL GAINS AT THE ACCURACY FRONTIER. **The best of all $\binom{5}{3}$=10 three-judge juries (Phi-4 + Qwen3-14B + Phi-4-Reasoning-Plus) reaches 89.61%, just 0.06% above the best individual judge.** Drawing panels from the top-5 judges, majority voting reduces error only when members err independently: for a K-judge panel with per-member error ϵ and pairwise error correlation $\bar{\rho}$, the voted-error variance scales as $\frac{1}{K}\epsilon(1-\epsilon)\,[1+(K-1)\bar{\rho}]$, which stops shrinking as $\bar{\rho} \to 1$. At the frontier the top judges share most mistakes, so all ten juries collapse into the band 88.82–89.61%; constructing maximally decorrelated panels (e.g., combining models with orthogonal error profiles) might yield larger gains but requires per-instance error knowledge unavailable at deployment time (Appendix B).

MULTI-AGENT DEBATE DEGRADES BINARY JUDGING. **Under the RCR protocol, no debate**

beats the individual baseline of its strongest member. The best debate (Qwen3-4B Variant B, 87.04%) falls 0.77% below Qwen3-4B's individual 87.81% at B=8,192; Phi-4-mini Variant B (86.92%) loses 0.69%; and the best cross-architecture Variant A (84.52%) loses 3.29% relative to its strongest constituent. We interpret this as a property of the task: a binary verdict carries little structured content for agents to exchange, so the RCR "critique" step can transmit a confident wrong verdict into a previously correct agent rather than correct it (qualitative analysis in Appendix D), echoing Wynn et al. (2025) and Zhang et al. (2025a). This contrasts with open-ended settings where debate can help (Chan et al., 2024); the result is specific to RCR and binary judging, and other aggregation rules may behave differently.

Practical implication. For binary correctness judging, a single strong judge (Phi-4, 89.55%) is simultaneously more accurate and cheaper, by a factor of three in forward passes, than any ensemble or debate we evaluated.

4.5 Open-Ended Evaluation

CLOSED-ENDED AND OPEN-ENDED RANKINGS DIVERGE. **The best binary judge, Phi-4 (rank 1 closed-ended), falls to rank 9 on MT-Bench (ρ=0.21), while always-thinking models invert the ordering.** Scoring all 16 judges on SummEval (1,600 pairs, SLM-Human correlation) and MT-Bench (80 questions, SLM-LLM correlation), closed-ended accuracy predicts SummEval agreement strongly (r=0.84 with Spearman ρ) but MT-Bench agreement only moderately (r=0.61; full results in Appendix C). We hypothesize that closed-ended judging primarily requires answer verification against a reference, while open-ended scoring needs an overall quality assessment (coherence, depth, style) that benefits from extended reasoning; the two paradigms thus rely on overlapping but distinct skills.

REASONING MODELS EXCEL AT OPEN-ENDED SCORING. **Phi-4-Reasoning reaches ρ=0.57 on MT-Bench (SLM-LLM), 2.7× its non-thinking sibling Phi-4 (ρ=0.21).** The Phi-4 family gives a near-controlled comparison: the two share a base model and differ chiefly in always-thinking behavior (alongside the reasoning-oriented post-training that enables it), yet non-thinking Phi-4 leads on closed-ended accuracy and SLM-Human correlation while always-thinking Phi-4-Reasoning leads on SLM-LLM correlation. A natural reading is that reasoning tokens act as *signal* for graded multi-turn quality but as *noise* for binary right/wrong calls, the domain-dependence of §4.2 re-expressed in the open-ended setting.

FLUENCY IS THE HARDEST DIMENSION FOR SLM JUDGES. On SummEval (SLM-Human), top judges reach ρ=0.53–0.62 on coherence but only 0.36–0.42 on fluency: SLM judges track *logical* structure (coherence, consistency) better than *surface* quality (grammar, readability), suggesting that fluency scoring in particular still benefits from human cross-validation.

THE QUICK-VERDICT JUDGE CLOSES THE COST GAP. **The best SLM judge reaches 89.55% at B=10, shrinking the output term βB of the cost C_{total} by nearly three orders of magnitude versus a reasoned judge.** Since the shared prompt makes the verdict cost output-dominated, the quick-to-reasoned ratio is $\frac{\alpha B_{\text{in}}+10\beta}{\alpha B_{\text{in}}+8192\beta} \to \frac{10}{8192} \approx 1.2\times10^{-3}$. Because $\Delta_{\text{math}} > 0$ for most judges (§4.2), this near-800× saving on math judging typically comes *with* a small accuracy gain rather than a loss, making B=10 an attractive default where correctness is verifiable; on general tasks the same budget trades some accuracy for cost, so the optimal budget is itself domain-dependent.

5 Conclusion and Future Work

We introduce SLMJURY, a framework that casts SLM judging as a budget-conditioned function and evaluates it across closed- and open-ended paradigms over 16 judges, 10 benchmarks, and 3,900+ experiments. Four findings stand out: (1) the overthinking effect is *domain-dependent*, its sign set by task structure rather than model capability; (2) domain generalization separates families, with math-to-general gaps from under 10% to nearly 40%; (3) the two paradigms rely on different capabilities, inverting the judge ranking; and (4) under the RCR protocol, debate degrades accuracy across all tested configurations, while the top judges resist six adversarial personas with at most 0.55% variance. Reliable automated evaluation does not require large proprietary models, yet no single SLM excels across all paradigms. The practical recipe is therefore not a bigger judge but a *budget-matched* one: a quick-verdict SLM is best for verifiable tasks, and a reasoning-trained judge for open-ended scoring. Promising next steps are an adaptive budget policy $B^\star(\mathcal{D})$, judge-specific fine-tuning, and error-decorrelated panel selection.

Limitations

We acknowledge several limitations. **First**, our closed-ended evaluation relies on a programmatic oracle for ground truth, so tasks without clear correct answers (e.g., creative writing) are outside our scope; the oracle's equivalence function, though multi-strategy (numeric, symbolic, label matching; Appendix G), may still misclassify edge cases in LaTeX or numeric formatting for a small fraction of problems. **Second**, open-ended evaluation relies on human annotations (SummEval) and large oracle model scores via the Together API (MT-Bench), both of which introduce ground-truth noise; oracle scores vary across models, and averaging over two oracles may not fully remove this variance. **Third**, while we span three closed-ended domains and two open-ended benchmarks, our findings may not generalize to subjective evaluation or specialized domains such as medical or legal reasoning, and we omit proprietary baselines (GPT-4) due to cost and reproducibility constraints. **Fourth**, results are averaged over two student models chosen to span the capability spectrum (95.45% vs. 82.49% on GSM8K); other student distributions may yield different findings. **Fifth**, debate uses the RCR protocol of Srivastava et al. (2025a), and other aggregation rules may behave differently. **Sixth**, we study reference-based verification, the standard LLM-as-a-judge paradigm; reference-free judging is a distinct, harder task we leave for future work, as is a sweep over intermediate token budgets (e.g., 64, 256, 1024, 4096) to test whether the overthinking effect is gradual or threshold-like. **Finally**, persona robustness is tested with six English-language personas and may not transfer to other languages or phrasings. Full reproducibility details appear in Appendix P.

Potential Risks. Our work does not pose direct risks, but reliance on automated SLM judges in critical applications (e.g., education, hiring) should consider robustness issues. Our persona experiments show that adversarial system prompts can degrade weaker models by over 10%, highlighting that deployment without robustness testing could produce unreliable evaluations. To enable transparent benchmarking, the framework will be released publicly upon publication, and we encourage further research on judge calibration.

Ethical Considerations

This study evaluates small language models using standardized benchmarks and publicly available datasets, ensuring transparency and reproducibility. No private or sensitive data was used, and all models were assessed under fair and consistent conditions. We acknowledge potential biases in both programmatic and LLM-based evaluations and encourage further research for mitigation. All model and dataset licenses are documented in Appendix P.

AI Assistance. We used AI assistants for parts of the implementation and manuscript preparation, including generating LaTeX code for tables and refining text written by the authors. All AI-generated content was carefully reviewed and revised by the authors to ensure accuracy and clarity.

References

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, and 1 others. 2025. Phi-4-reasoning technical report. *arXiv preprint arXiv:2504.21318*.

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Tassallah Abdullahi, Shrestha Ghosh, Hamish S Fraser, Daniel León Tramontini, Adeel Abbasi, Ghada Bourjeily, Carsten Eickhoff, and Ritambhara Singh. 2026. The persona paradox: Medical personas as behavioral priors in clinical language models. *arXiv preprint arXiv:2601.05376*.

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, and 1 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Pranjal Aggarwal, Seungone Kim, Jack Lanchantin, Sean Welleck, Jason Weston, Ilia Kulikov, and Swarnadeep Saha. 2025. Optimalthinkingbench: Evaluating over and underthinking in llms. *arXiv preprint arXiv:2508.13141*.

Firoj Alam, Gagan Bhatia, Sahinur Rahman Laskar, and Shammur Absar Chowdhury. 2026. Beyond llm-as-a-judge: Deterministic metrics for multilingual generative text evaluation. *arXiv preprint arXiv:2604.05083*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Zhenyu Bi, Gaurav Srivastava, Yang Li, Swastik Roy, Meng Lu, Morteza Ziyadi, and Xuan Wang. 2026. Judgeboard: Benchmarking and enhancing small language models for reasoning evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30076–30084.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. 2024. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. Chateval: Towards better llm-based evaluators through multi-agent debate. In *International conference on learning representations*, volume 2024, pages 9079–9093.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or llms as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327.

Hyunjoon Cho, Sungmin Kang, Gabin An, and Shin Yoo. 2025. Cosmosfl: Ensemble of small language models for fault localisation. In *2025 IEEE/ACM International Workshop on Large Language Models for Code (LLM4Code)*, pages 17–24. IEEE.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Giuseppe Crupi, Rosalia Tufano, and Gabriele Bavota. 2026. Improving code generation via small language model-as-a-judge. *arXiv preprint arXiv:2602.11911*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first international conference on machine learning*.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Soumya Suvra Ghosal, Souradip Chakraborty, Avinash Reddy, Yifu Lu, Mengdi Wang, Dinesh Manocha, Furong Huang, Mohammad Ghavamzadeh, and Amrit Singh Bedi. 2026. Does thinking more always help? mirage of test-time scaling in reasoning models. *Advances in Neural Information Processing Systems*, 38:172664–172691.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *The Innovation*.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. 2025. Token-budget-aware llm reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24842–24855.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Tianyu Hu, Zhen Tan, Song Wang, Huaizhi Qu, and Tianlong Chen. 2026. Multi-agent debate for llm judges with adaptive stability detection. *Advances in Neural Information Processing Systems*, 38:46504–46540.

Pratik Jayarao, Himanshu Gupta, Neeraj Varshney, and Chaitanya Dwivedi. 2025. Explicit reasoning makes better judges: A systematic study on accuracy, efficiency, and robustness. *arXiv preprint arXiv:2509.13332*.

Hongchao Jiang, Yiming Chen, Yushi Cao, Hung-yi Lee, and Robby T Tan. 2025. Codejudgebench: Benchmarking llm-as-a-judge for coding tasks. *arXiv preprint arXiv:2507.10535*.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient

memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, and 1 others. 2025. From generation to judgment: Opportunities and challenges of llm-as-a-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2757–2791.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2961–2984.

Zhuochun Li, Yong Zhang, Ming Li, Yuelyu Ji, Yiming Zeng, Ning Cheng, Yun Zhu, Yanmeng Wang, Shaojun Wang, Jing Xiao, and 1 others. 2026. Rethinking llm-as-a-judge: Representation-as-a-judge with small language models via semantic capacity asymmetry. *arXiv preprint arXiv:2601.22588*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 17889–17904.

Dachuan Lin, Guobin Shen, Zihao Yang, Tianrong Liu, Dongcheng Zhao, and Yi Zeng. 2025. Efficient llm safety evaluation through multi-agent debate. *arXiv preprint arXiv:2511.06396*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 3214–3252.

Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. 2025. Reasoning models can be effective without thinking. *arXiv preprint arXiv:2504.09858*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Michael J Ryan, Omar Shaikh, Aditri Bhagirath, Daniel Frees, William Barr Held, and Diyi Yang. 2025. Synthesizeme! inducing persona-guided prompts for personalized reward models in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8045–8078.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Leonard Salewski, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata. 2023. In-context impersonation reveals large language models' strengths and biases. *Advances in neural information processing systems*, 36:72044–72057.

Parshin Shojaee, Iman Mirzadeh, Maxwell Horton, Samy Bengio, Mehrdad Farajtabar, and 1 others. 2026. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Advances in Neural Information Processing Systems*, 38:108018–108059.

Gaurav Srivastava, Zhenyu Bi, Meng Lu, and Xuan Wang. 2025a. Debate, train, evolve: Self-evolution of language model reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32752–32798.

Gaurav Srivastava, Shuxiang Cao, and Xuan Wang. 2025b. Thinkslm: Towards reasoning in small language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32600–32650.

Gaurav Srivastava, Aafiya Hussain, Zhenyu Bi, Swastik Roy, Priya Pitre, Meng Lu, Morteza Ziyadi, and Xuan Wang. 2025c. Beyondbench: Benchmark-free evaluation of reasoning in language models. *arXiv preprint arXiv:2509.24210*.

Gaurav Srivastava, Aafiya Hussain, Sriram Srinivasan, and Xuan Wang. 2025d. Do llms overthink basic math reasoning? benchmarking the accuracy-efficiency tradeoff in language models. *arXiv preprint arXiv:2507.04023*.

Gaurav Srivastava, Aafiya Hussain, Chi Wang, Yingyan Celine Lin, and Xuan Wang. 2026. Effgen: Enabling small language models as capable autonomous agents. *arXiv preprint arXiv:2602.00887*.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, and 1 others. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Annalisa Szymanski, Noah Ziems, Heather A Eicher-Miller, Toby Jia-Jun Li, Meng Jiang, and Ronald A Metoyer. 2025. Limitations of the llm-as-a-judge

approach for evaluating llm outputs in expert knowledge tasks. In *Proceedings of the 30th international conference on intelligent user interfaces*, pages 952–966.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Tang, Alejandro Cuadron, Chenguang Wang, Raluca Popa, and Ion Stoica. 2025. Judgebench: A benchmark for evaluating llm-based judges. In *International Conference on Learning Representations*, volume 2025, pages 63277–63303.

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in llms: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16612–16631.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating llm generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.

Junlin Wang, Siddhartha Jain, Dejiao Zhang, Baishakhi Ray, Varun Kumar, and Ben Athiwaratkun. 2024a. Reasoning in token economies: budget-aware evaluation of llm reasoning strategies. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19916–19939.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yidong Wang, Zhuohao Yu, Wenjin Yao, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, and 1 others. 2024b. Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization. In *International Conference on Learning Representations*, volume 2024, pages 43573–43593.

Yike Wang, Faeze Brahman, Shangbin Feng, Teng Xiao, Hannaneh Hajishirzi, and Yulia Tsvetkov. 2026. Small reward models via backward inference. *arXiv preprint arXiv:2602.13551*.

Andrea Wynn, Harsh Satija, and Gillian Hadfield. 2025. Talk isn't always cheap: Understanding failure modes in multi-agent debate. *arXiv preprint arXiv:2509.05396*.

Austin Xu, Srijan Bansal, Yifei Ming, Semih Yavuz, and Shafiq Joty. 2025a. Does context matter? contextualjudgebench for evaluating llm-based judges in contextual settings. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9541–9564.

Austin Xu, Yilun Zhou, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. 2025b. J4r: Learning to judge with equivalent initial state group relative policy optimization. *arXiv preprint arXiv:2505.13346*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Haoyan Yang, Runxue Bao, Cao Danica Xiao, Jun Ma, Parminder Bhatia, Shangqian Gao, and Taha Kass-Hout. 2026. Any large language model can be a reliable judge: Debiasing with a reasoning-based bias detector. *Advances in Neural Information Processing Systems*, 38:6318–6362.

Hanling Yi, Feng Lin, Hongbin Li, Ning Peiyang, Xiaotian Yu, and Rong Xiao. 2024. Generation meets verification: Accelerating large language model inference with smart parallel auto-correct decoding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5285–5299.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800.

Hangfan Zhang, Zhiyao Cui, Jianhao Chen, Xinrun Wang, Qiaosheng Zhang, Zhen Wang, Dinghao Wu, and Shuyue Hu. 2025a. Stop overvaluing multi-agent debate–we must rethink evaluation and embrace model heterogeneity. *arXiv preprint arXiv:2502.08788*.

Taolin Zhang, Maosong Cao, Alexander Lam, Songyang Zhang, and Kai Chen. 2025b. Compassjudger-2: Towards generalist judge model via verifiable rewards. *arXiv preprint arXiv:2507.09104*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. Judgelm: Fine-tuned large language models are scalable judges. In *International Conference on Learning Representations*, volume 2025, pages 51257–51296.

Appendix

This appendix provides supplementary material organized in three parts: **(A)** Extended results including per-dataset breakdowns, full ensemble and multi-agent debate tables, open-ended evaluation results, and error analysis; **(B)** Methodology and algorithms including the formal framework, answer extraction and normalization, open-ended scoring metrics, debate and ensemble algorithms, student solver details, and implementation details; **(C)** Prompt templates, persona prompts, dataset details, model configurations, and reproducibility information.

Contents of the Appendix

A Per-Dataset Detailed Results **13**

B Full Ensemble and Debate Results **14**
- B.1 Majority-Voting Ensembles 14
- B.2 Multi-Agent Debate 14

C Full Open-Ended Evaluation Results **14**
- C.1 SummEval: Full SLM-Human Correlation Results 15
- C.2 SummEval: Per-Dimension Breakdown 15
- C.3 MT-Bench: Full SLM-LLM Correlation Results 15

D Error Analysis **16**
- D.1 Error Distribution by Verdict Type 17
- D.2 Overthinking Error Patterns 17
- D.3 Debate-Induced Errors 18
- D.4 Persona-Induced Error Patterns . . 18

E Formal Framework **19**
- E.1 Spaces and Objects 19
- E.2 The Judge Function 20
- E.3 The Programmatic Oracle 20
- E.4 Closed-Ended Metrics 20

F Answer Extraction **20**

G Answer Normalization and Equivalence **21**
- G.1 Type-Based Dispatch 21
- G.2 Numeric Normalization 21
- G.3 LaTeX Multi-Strategy Comparison 21
- G.4 Verdict Parsing 22

H Open-Ended Scoring Metrics and Parsing **22**
- H.1 Score Vectors 22
- H.2 Correlation and Agreement Metrics 22
- H.3 Score Parsing 22

I Collaborative Judging: Algorithms and Prompts **23**
- I.1 Majority-Voting Ensemble 23
- I.2 RCR Multi-Agent Debate 23
- I.3 Debate Configurations 23
- I.4 RCR Prompt: Math Datasets . . . 24
- I.5 RCR Prompt: Science Datasets . . 24
- I.6 RCR Prompt: General Reasoning Datasets 24

J Student Solution Generation **28**
- J.1 Overview 28
- J.2 Generation Process 28
- J.3 Student Model Performance . . . 28
- J.4 Why Two Student Models? 28
- J.5 Oracle Extraction Reliability . . . 28

K Implementation Details **28**
- K.1 Hardware 28
- K.2 Software Stack 29
- K.3 Inference Configuration 29
- K.4 Compute Budget 29

L Prompt Templates **30**
- L.1 Judge Prompt: Quick Verdict (10 Tokens) 30
- L.2 Judge Prompt: Reasoned Verdict (8,192 Tokens) 30
- L.3 Student Solver Prompts 30
- L.4 Open-Ended Scoring Prompts . . 30

M Persona System Prompts **35**

N Dataset Details **38**

O Model Configurations **39**

P Reproducibility and Licensing **39**
- P.1 Code and Data Availability 39
- P.2 Reproducibility Checklist 39
- P.3 Model Licenses 39
- P.4 Dataset Licenses 39

A Per-Dataset Detailed Results

Tables 5 and 6 present accuracy broken down by individual dataset for all judge models at each token setting, and Figure 6 contrasts the two budgets per

judge. These results complement the aggregate accuracy in Table 2 (main paper), which averages over all datasets and both student models.

Key Observations. GSM8K is the easiest benchmark: most models above 3B exceed 95% at either token setting, and at 10 tokens the top-5 all clear 97.5%, with only a 1.02% gap between Qwen3-14B (98.90%) and LLaMA-3.1-8B (97.88%), a clear ceiling effect. **HellaSwag is the hardest**: even the best judges reach only 79.52% (Phi-4) and 79.43% (Qwen3-8B) at 10 tokens, far below their 92–98% on math, because its sentence-completion format needs commonsense that brief verdicts cannot capture. **WinoGrande shows the largest overthinking swing**: Qwen3-4B gains 23.12% from reasoning (71.43% $\rightarrow$ 94.55%), the largest single-dataset gain in the study, since pronoun resolution demands step-by-step disambiguation. **Qwen 2.5 models are extreme math specialists**: Qwen2.5-7B scores 97.50% on GSM8K but only 55.35% on HellaSwag (a 42.15% gap), and Qwen2.5-3B follows the same pattern (96.06% vs. 55.75%), reflecting heavily math-focused instruction tuning. **Reasoning helps most on general and science tasks**: LLaMA-3.2-1B gains 32.45% on ARC-Easy (48.02% $\rightarrow$ 80.47%) and 21.90% on WinoGrande (48.62% $\rightarrow$ 70.52%), and for capable models like Qwen3-4B general tasks gain 5–23% while math tasks lose 1–6%. **Math overthinking is real**: Qwen2.5-3B drops 10.51% on GSM-Plus (96.01% $\rightarrow$ 85.50%) and 11.77% on MATH (91.29% $\rightarrow$ 79.52%) under reasoning, where quick judgment was already reliable.

Cross-Student Analysis. An important dimension not visible in the per-dataset averages is how judge accuracy varies across student models. Qwen2.5-32B produces higher-quality solutions (95.45% accuracy on GSM8K) than LLaMA-3.1-8B (82.49%). Judges tend to be more accurate when evaluating the stronger student's solutions, because correct solutions are easier to verify than incorrect ones. The accuracy gap between students is smallest for the strongest judges (Qwen3-14B: $<$1% difference) and largest for the weakest (LLaMA-3.2-1B: 5–8% difference), suggesting that stronger judges generalize better across solution quality levels.

B Full Ensemble and Debate Results

This section reports the complete results for the two collaborative judging strategies summarized in §4.4: majority-voting ensembles (Table 7) and multi-agent debate (Table 8). The protocols, prompts, and the RCR algorithm are deferred to Appendix I; here we focus on the numbers and what they reveal. Figure 7 places both strategies against the best individual judge on a single axis.

B.1 Majority-Voting Ensembles

Table 7 lists all $\binom{5}{3}$=10 three-judge panels drawn from the top-5 individual judges. For each problem, the panel verdict is the majority label V_{maj} defined in Eq. 13 (Appendix I). The best panel (Phi-4 + Qwen3-14B + Phi-4-R-Plus) reaches 89.61%, just +0.06% over the best individual judge, and all ten panels lie inside the narrow band 88.82–89.61%. The reason is structural: at the accuracy frontier the constituent judges share most of their errors, so the pairwise error correlation $\bar{\rho}$ approaches 1 and the variance-reduction term $\frac{1}{K}\epsilon(1-\epsilon)[1+(K-1)\bar{\rho}]$ stops shrinking (§4.4).

B.2 Multi-Agent Debate

Table 8 reports the accuracy of the 7 RCR debate configurations that ran at the standard token budget (B=8,192): two cross-architecture Variant A and five intra-model Variant B (Appendix I.3). The best debate (Qwen3-4B Variant B, 87.04%) trails the best individual judge by 2.51% and trails its own constituent (Qwen3-4B at B=8,192, 87.81%) by 0.77%. Both Variant A configurations underperform the individual accuracy of their strongest member, confirming that architectural diversity does not offset the persuasion vulnerability analyzed in Appendix D.

Reduced-budget Variant A configurations. Three additional cross-architecture Variant A configurations required a reduced token budget of B=4,096 because their multi-round prompts exceeded the maximum context length at B=8,192. We report these separately in Table 9 to avoid confounding token-budget effects with architectural effects. These results are excluded from all claims and comparisons in the main paper.

C Full Open-Ended Evaluation Results

This section presents the complete open-ended evaluation results for all 16 SLM judges. SummEval measures SLM-Human correlation (SLM

		Math			Science		General			
Family	**Judge Model**	**GSM8K**	**GSM+**	**MATH**	**ARC-E**	**ARC-C**	**HSwag**	**WGrnd**	**TQA**	**Avg**
LLaMA 3.x	LLaMA-3.2-1B	31.54	39.61	42.78	48.02	45.86	46.19	48.62	46.71	43.15
	LLaMA-3.2-3B	94.43	84.03	75.46	85.90	83.66	72.62	88.36	77.78	79.76
	LLaMA-3.1-8B	97.88	93.43	91.13	86.87	85.28	76.03	91.55	82.68	86.79
Qwen 2.5	Qwen2.5-1.5B	73.62	76.94	84.41	78.43	72.14	67.26	63.50	55.56	73.92
	Qwen2.5-3B	96.06	96.01	91.29	69.00	68.69	55.75	58.52	61.55	77.65
	Qwen2.5-7B	97.50	97.16	93.12	60.84	59.98	55.35	64.40	56.36	77.45
Qwen 3	Qwen3-0.6B	89.20	72.07	65.48	85.06	80.50	56.48	72.77	58.55	67.92
	Qwen3-1.7B	95.79	88.52	83.68	87.31	86.69	77.19	85.87	79.61	84.11
	Qwen3-4B	98.33	97.74	93.80	83.08	83.92	76.62	71.43	76.46	87.56
	Qwen3-8B	98.45	94.45	94.01	87.82	87.93	79.43	93.84	85.89	88.96
	Qwen3-14B	98.90	96.20	94.13	87.86	87.93	79.36	93.17	84.94	89.51
Phi-4	Phi-4	98.45	96.98	92.12	88.01	88.10	79.52	93.92	86.04	89.55
	Phi-4-mini	98.22	95.33	93.33	87.77	87.16	77.13	91.75	81.36	88.22

Table 5: **Per-dataset accuracy (%) at 10 tokens** (quick verdict). Avg = arithmetic mean over all 8 datasets, averaged across 2 student models. Only the 13 dual-setting models are shown (excludes always-thinking models).

		Math			Science		General			
Family	**Judge Model**	**GSM8K**	**GSM+**	**MATH**	**ARC-E**	**ARC-C**	**HSwag**	**WGrnd**	**TQA**	**Avg**
LLaMA 3.x	LLaMA-3.2-1B	74.91	67.47	58.55	80.47	77.69	60.60	70.52	58.92	65.53
	LLaMA-3.2-3B	91.51	89.10	84.36	84.70	81.61	70.82	83.19	79.02	81.77
	LLaMA-3.1-8B	96.82	89.67	89.44	85.77	83.79	73.81	75.49	77.27	83.70
Qwen 2.5	Qwen2.5-1.5B	90.11	88.50	89.08	85.54	81.61	64.78	78.30	63.01	79.91
	Qwen2.5-3B	92.91	85.50	79.52	79.95	78.46	65.53	65.67	68.27	76.89
	Qwen2.5-7B	97.95	94.49	90.87	85.94	84.85	75.69	85.44	80.34	86.62
Qwen 3	Qwen3-0.6B	95.07	83.95	81.08	80.51	76.15	62.65	80.43	68.93	76.37
	Qwen3-1.7B	96.40	88.88	90.32	87.86	87.24	78.41	90.33	82.82	85.96
	Qwen3-4B	97.92	91.59	92.66	87.79	88.01	79.39	94.55	85.45	87.81
	Qwen3-8B	97.46	90.04	93.22	87.92	88.10	79.58	94.32	85.16	87.43
	Qwen3-14B	96.47	91.55	93.48	87.29	87.88	79.69	94.67	85.89	87.93
Phi-4	Phi-4	97.54	90.32	91.89	88.03	88.14	79.24	92.70	85.53	87.17
	Phi-4-R†	98.26	91.94	93.81	87.98	88.27	79.72	94.55	85.53	88.24
	Phi-4-R-Plus†	98.41	93.31	94.02	87.94	88.18	79.73	94.95	85.89	88.75
	Phi-4-mini	98.33	96.18	90.85	87.67	87.46	75.63	90.06	82.53	87.61
	Phi-4-mini-R†	95.98	89.65	78.51	86.17	84.39	77.59	77.55	79.46	83.32

Table 6: **Per-dataset accuracy (%) at 8,192 tokens** (reasoned verdict). † = always-thinking models (8,192 tokens only). All 16 models included.

scores vs. human expert annotations on machine-generated summaries), while MT-Bench measures SLM-LLM correlation (SLM scores vs. large oracle model scores on multi-turn conversations). Table 4 in the main paper reports only the top-5 models; Tables 10 and 12 provide the full rankings.

C.1 SummEval: Full SLM-Human Correlation Results

Table 10 reports all five agreement metrics (Pearson r, Spearman ρ, Cohen's κ, accuracy, MSE) between each judge's scores and the human expert annotations, averaged over the four SummEval dimensions. The Phi-4 and Qwen 3 families dominate, while the smallest judges fall to near-zero correlation, mirroring the closed-ended capability ordering.

C.2 SummEval: Per-Dimension Breakdown

Table 11 shows SLM-Human Spearman correlation broken down by the four SummEval dimensions for the top-8 judges. Consistency is the easiest dimension for most models, while fluency is consistently the hardest.

C.3 MT-Bench: Full SLM-LLM Correlation Results

Table 12 reports the same five metrics against the large oracle scores, averaged over the four oracle-student combinations. Here the ordering shifts: the always-thinking reasoning models rise to the top, since their graded scoring better matches how the

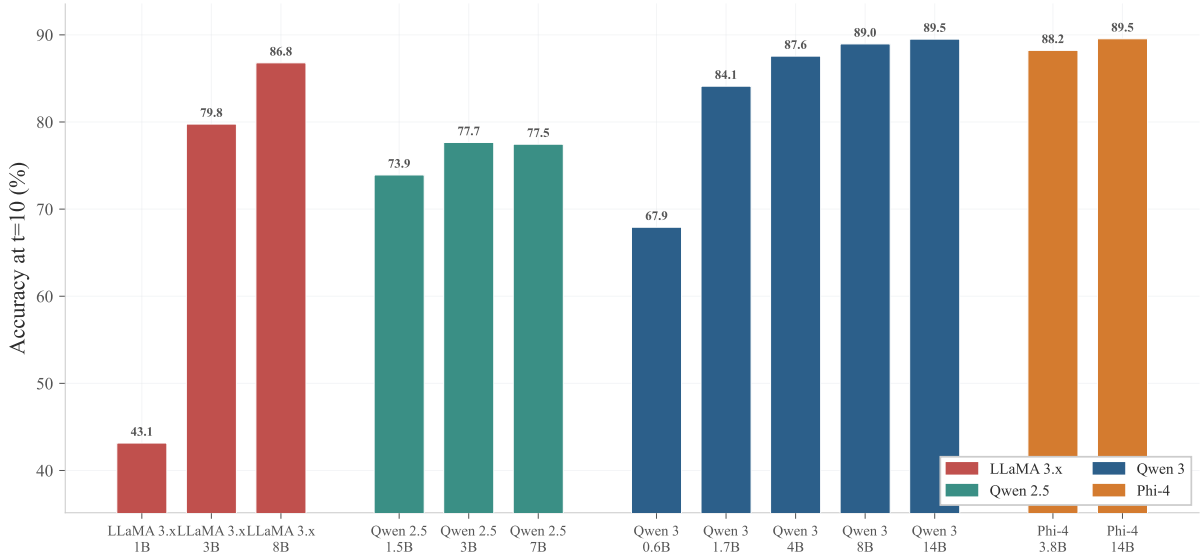

Figure 5: **Judge accuracy by model family and size** at 10 tokens. Within each family, accuracy generally scales with parameters, but domain generalization varies significantly across families. Phi-4 and Qwen 3 maintain the most uniform accuracy across all eight benchmarks.

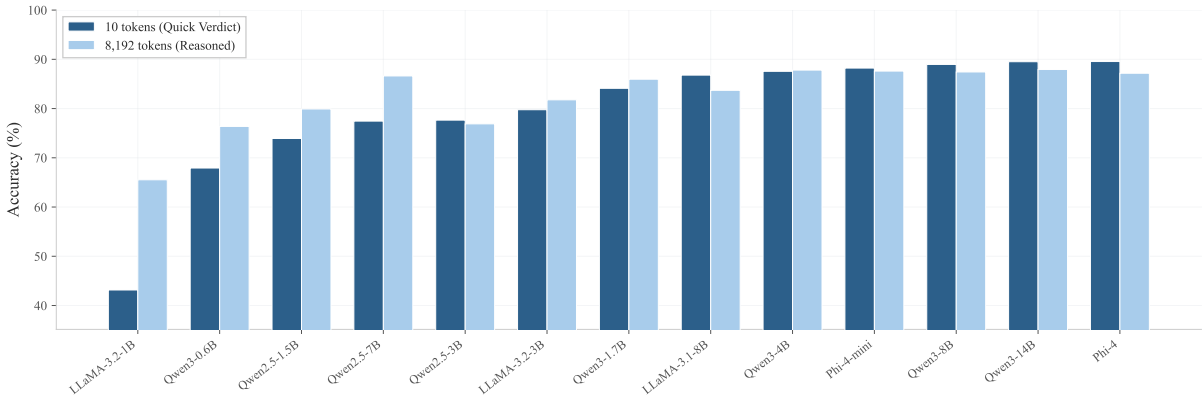

Figure 6: **Token budget comparison** for all 13 dual-setting judges, showing absolute accuracy at both B=10 and B=8,192. The mixed pattern of which setting wins reflects opposing domain effects: quick verdicts dominate on math while reasoning dominates on general tasks (see §4.2).

oracles rate multi-turn conversations.

Key Observations. SLM-Human and SLM-LLM rankings diverge: Phi-4 (rank 1 on SummEval SLM-Human, ρ=0.520) drops to rank 9 on MT-Bench SLM-LLM (ρ=0.215), while Phi-4-Reasoning (rank 4 SLM-Human) rises to rank 1 SLM-LLM (ρ=0.570), showing that the two correlation regimes demand different capabilities. **Reasoning models achieve higher SLM-LLM correlation**: the three always-thinking models (Phi-4-Reasoning, Phi-4-Reasoning-Plus, Phi-4-mini-Reasoning) all rank in the top-6 on MT-Bench, versus ranks 4, 2, and 8 on SummEval, because extended reasoning better approximates large oracle scoring on multi-turn conversations. **Sub-1B models produce near-random correlations**: LLaMA-3.2-1B and Qwen3-0.6B both score Spearman $\rho < 0.08$ on both benchmarks, marking a minimum capability threshold for meaningful open-ended evaluation. **Accuracy and correlation can diverge**: Qwen2.5-1.5B reaches 70.00% accuracy on MT-Bench yet $\rho = 0.018$, assigning similar scores to most responses without discriminating quality.

D Error Analysis

This section provides a deeper analysis of judge errors, examining when and why judges produce incorrect verdicts.

Judge 1	Judge 2	Judge 3	Acc (%)	IFR (%)
Phi-4 (t10)	Qwen3-14B (t10)	Phi-4-R-Plus (t8192)	**89.61**	100.00
Phi-4 (t10)	Qwen3-14B (t10)	Phi-4-R (t8192)	89.56	99.99
Phi-4 (t10)	Qwen3-14B (t10)	Qwen3-8B (t10)	89.56	100.00
Phi-4 (t10)	Qwen3-8B (t10)	Phi-4-R-Plus (t8192)	89.30	100.00
Qwen3-14B (t10)	Qwen3-8B (t10)	Phi-4-R-Plus (t8192)	89.27	100.00
Phi-4 (t10)	Qwen3-8B (t10)	Phi-4-R (t8192)	89.18	100.00
Qwen3-14B (t10)	Qwen3-8B (t10)	Phi-4-R (t8192)	89.18	100.00
Qwen3-14B (t10)	Phi-4-R-Plus (t8192)	Phi-4-R (t8192)	88.91	100.00
Phi-4 (t10)	Phi-4-R-Plus (t8192)	Phi-4-R (t8192)	88.87	100.00
Qwen3-8B (t10)	Phi-4-R-Plus (t8192)	Phi-4-R (t8192)	88.82	100.00

Table 7: **Full majority-voting results.** All 10 three-judge panels from the top-5 individual judges. The best ensemble (89.61%) provides only +0.06% over the best individual judge (Phi-4, 89.55%), evidence that the top judges already share highly correlated error patterns.

Variant	Agent 1	Agent 2	Agent 3	Acc (%)	IFR (%)
A	Qwen3-4B (t0)	Qwen2.5-3B (t0)	Qwen3-1.7B (t0)	84.52	99.98
	LLaMA-3.2-3B (t0)	Qwen2.5-1.5B (t0)	Qwen3-1.7B (t0)	83.89	99.88
B	Qwen3-4B ×3 (t0/0.4/0.9)			**87.04**	100.00
	Phi-4-mini ×3 (t0/0.4/0.9)			86.92	99.83
	Qwen2.5-1.5B ×3 (t0/0.4/0.9)			78.01	99.66
	Qwen3-0.6B ×3 (t0/0.4/0.9)			75.59	98.40
	LLaMA-3.2-1B ×3 (t0/0.4/0.9)			62.96	98.31

Table 8: **Full debate results (B=8,192).** All 7 MAD configurations that ran at the standard token budget. The best debate (Qwen3-4B Variant B, 87.04%) falls 2.51% below the best individual judge (89.55%). Both Variant A configurations underperform the individual accuracy of its strongest constituent at B=8,192 (Qwen3-4B at 87.81%).

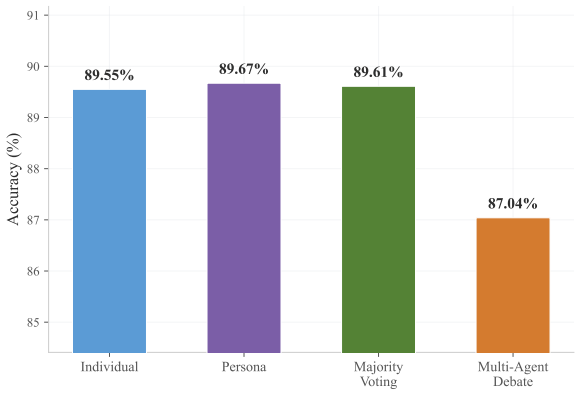

Figure 7: **Comparison of closed-ended evaluation approaches.** Majority voting yields a marginal gain (+0.06%) over the best individual judge (89.55%), while every debate configuration falls 0.69–3.29% below the individual baseline of its strongest constituent.

D.1 Error Distribution by Verdict Type

Judge errors fall into two categories: *false positives* (judging an incorrect solution as correct) and *false negatives* (judging a correct solution as incorrect). For the top-3 judges (Phi-4, Qwen3-14B, Qwen3-8B) at B=10, **false positives dominate**: roughly 60–65% of errors accept an incorrect answer rather than reject a correct one, consistent with the "default accept" bias documented in prior LLM-as-a-judge work. The remaining **false negatives cluster on MATH and general reasoning**: on competition-level math the judge occasionally flags correct solutions that use uncommon notation or non-standard derivations (which the SymPy oracle still resolves), while on general tasks (especially WinoGrande and TruthfulQA) they arise when the judge misreads the context-dependent nature of the correct answer.

D.2 Overthinking Error Patterns

The overthinking effect (where extended reasoning degrades accuracy) produces domain-dependent error patterns. For Qwen2.5-3B on math, which shows the largest overthinking penalty, errors at B=10 are sparse and evenly distributed (96.06% on GSM8K, 96.01% on GSM-Plus). At B=8,192 accuracy drops to 92.91% and 85.50% respectively (−10.51% on GSM-Plus), and the errors concentrate on problems where the student's reasoning is partially correct but the final answer is wrong: the judge's extended reasoning follows the student's chain of thought, gets led astray by plausible intermediate steps, and reaches an incorrect verdict. This is a form of reasoning sycophancy, where the judge defers to the student's process rather than

Variant	Agent 1	Agent 2	Agent 3	Acc (%)	IFR (%)
A (B=4,096)	LLaMA-3.2-3B (t0)	Qwen3-4B (t0)	Phi-4-mini (t0)	85.99	99.66
	Qwen2.5-1.5B (t0)	Qwen3-1.7B (t0)	Phi-4-mini (t0)	84.40	99.43
	Qwen3-4B (t0)	Phi-4-mini (t0)	Qwen2.5-3B (t0)	82.96	98.44

Table 9: **Reduced-budget debate results (B=4,096).** Three Variant A configurations where multi-round debate prompts exceeded the context window at B=8,192. These ran at B=4,096 and are reported separately; they are not used in any main-paper claims to ensure all debate comparisons use a consistent token budget.

Family	Judge Model	Pearson	Spearman	Cohen's κ	Accuracy	MSE
LLaMA 3.x	LLaMA-3.1-8B	0.380	0.293	0.240	72.81	0.954
	LLaMA-3.2-3B	0.101	0.078	0.034	65.76	1.412
	LLaMA-3.2-1B	−0.016	−0.013	−0.002	47.33	2.853
Qwen 2.5	Qwen2.5-7B	0.355	0.324	0.181	63.78	1.797
	Qwen2.5-3B	0.333	0.322	0.088	43.61	3.893
	Qwen2.5-1.5B	0.176	0.154	0.065	69.03	1.167
Qwen 3	Qwen3-14B	0.519	0.487	0.350	68.67	1.438
	Qwen3-8B	0.495	0.453	0.305	67.36	1.462
	Qwen3-4B	0.513	0.430	0.314	72.69	1.110
	Qwen3-1.7B	0.280	0.219	0.175	70.25	1.255
	Qwen3-0.6B	0.041	0.052	0.021	64.52	1.899
Phi-4	Phi-4	**0.574**	**0.520**	**0.399**	**74.29**	**1.258**
	Phi-4-Reasoning	0.528	0.479	0.355	67.56	1.840
	Phi-4-R-Plus	0.522	0.489	0.324	64.38	2.365
	Phi-4-mini	0.421	0.375	0.089	70.78	1.345
	Phi-4-mini-R	0.365	0.325	0.217	69.76	1.578

Table 10: **Full SummEval (SLM-Human) results for all 16 judges.** Correlations are computed between SLM judge scores and human expert annotations on machine-generated summaries across 4 dimensions (coherence, consistency, fluency, relevance). All judges use B=8,192. **Bold** = best per column.

independently checking the answer. Conversely, on general reasoning benchmarks, extended reasoning *helps*. Qwen3-4B gains 23.12% on WinoGrande (71.43% → 94.55%) because pronoun resolution requires step-by-step disambiguation that a 10-token verdict cannot provide. This bidirectional effect is the mechanistic basis for the domain-dependent overthinking thesis (§4.2).

D.3 Debate-Induced Errors

Multi-agent debate introduces a distinct failure mode: *correct-to-incorrect flips*. In the Qwen3-4B Variant B debate (which drops from 87.81% individual to 87.04% debate accuracy), problems where all three agents initially agree correctly stay correct, because consensus halts further rounds. The damage occurs when agents initially disagree: in 35–40% of these cases the correct agent is swayed by incorrect peer reasoning, as the RCR "Critique" step makes it doubt its own verdict against two opposing arguments. The temperature diversity of Variant B ($\tau \in \{0.0, 0.4, 0.9\}$) adds further stochastic flips, occasionally overturning a correct verdict even when the deterministic agent (τ=0.0) was right.

This pattern is amplified in Variant A cross-architecture debates, where weaker agents can corrupt the judgments of stronger ones. The best Variant A configuration at B=8,192 (Qwen3-4B + Qwen2.5-3B + Qwen3-1.7B at 84.52%) falls 3.29% below Qwen3-4B's individual accuracy at B=8,192 (87.81%), showing that architectural diversity does not compensate for the persuasion vulnerability.

D.4 Persona-Induced Error Patterns

The Lenient persona's effect on LLaMA-3.1-8B (−10.20%) follows a clear pattern: the model begins accepting incorrect solutions that have plausible reasoning chains. The Lenient system prompt instructs the judge to "give partial credit" and "not penalize minor errors," which the model interprets as lowering the threshold for correctness. This transforms the binary classification boundary from "is the answer correct?" to "is the reasoning attempt reasonable?," a fundamentally different and easier-to-satisfy criterion.

The Strict persona shows a larger effect than

Judge	Spearman ρ				Cohen's κ			
	Coher.	Consist.	Fluency	Relev.	Coher.	Consist.	Fluency	Relev.
Phi-4	**0.619**	**0.555**	**0.423**	**0.482**	**0.472**	**0.519**	0.223	**0.383**
Phi-4-R-Plus	0.583	0.501	0.382	0.492	0.424	0.452	0.048	0.373
Qwen3-14B	0.599	0.521	0.370	0.459	0.427	0.486	0.107	0.381
Phi-4-R	0.571	0.486	0.367	0.492	0.444	0.516	0.071	0.389
Qwen3-8B	0.534	0.478	0.357	0.443	0.329	0.408	0.132	0.352
Qwen3-4B	0.466	0.495	0.344	0.416	0.208	**0.534**	**0.287**	0.227
Phi-4-mini	0.405	0.342	0.372	0.381	0.019	0.244	0.032	0.059
Phi-4-mini-R	0.355	0.402	0.242	0.301	0.139	0.413	0.193	0.123

Table 11: **SummEval per-dimension SLM-Human correlations** for the top-8 judges (sorted by overall Spearman). Consistency receives the highest correlations overall; fluency is consistently the weakest dimension. **Bold** = best per column.

Family	Judge Model	Pearson	Spearman	Cohen's κ	Accuracy	MSE
LLaMA 3.x	LLaMA-3.1-8B	0.051	0.050	−0.018	68.75	2.200
	LLaMA-3.2-3B	0.072	0.055	0.027	69.37	1.713
	LLaMA-3.2-1B	0.113	0.074	0.129	69.38	1.769
Qwen 2.5	Qwen2.5-7B	0.196	0.216	−0.001	70.00	1.925
	Qwen2.5-3B	−0.031	−0.071	0.002	68.13	2.081
	Qwen2.5-1.5B	0.029	0.018	0.016	70.00	1.800
Qwen 3	Qwen3-14B	**0.522**	0.550	0.364	83.44	**1.003**
	Qwen3-8B	0.469	0.466	0.342	81.56	1.209
	Qwen3-4B	0.417	0.431	0.240	77.81	1.297
	Qwen3-1.7B	0.281	0.274	0.166	75.00	1.538
	Qwen3-0.6B	−0.045	−0.034	0.017	66.56	2.763
Phi-4	Phi-4	0.257	0.215	0.043	71.88	2.131
	Phi-4-Reasoning	0.508	**0.570**	0.319	81.25	1.206
	Phi-4-R-Plus	0.469	0.500	**0.434**	**84.06**	1.128
	Phi-4-mini	0.043	0.073	0.014	69.37	2.188
	Phi-4-mini-R	0.368	0.372	0.224	75.94	1.475

Table 12: **Full MT-Bench (SLM-LLM) results for all 16 judges.** Correlations are computed between SLM judge scores and large oracle model scores (averaged over 2 oracles $\times$ 2 students = 4 combinations, 80 questions each). All judges use B=8,192. **Bold** = best per column.

in math-only evaluation. Phi-4-mini at B=8,192 loses 11.91% under the Strict persona, the largest persona-induced drop in the study. At extended token budgets, the Strict framing causes the model to second-guess correct verdicts, producing false negatives. This contrasts with the near-zero Strict effect at B=10, where the quick verdict format leaves no room for doubt.

In contrast, the top judges (Qwen3-14B with 0.08% range, Phi-4 with 0.55% range at B=10) show robust invariance to all six personas, suggesting that well-trained models internalize a stable evaluation criterion that overrides surface-level framing.

E Formal Framework

This section gives the complete formal description of the SLMJURY pipeline that §3 summarizes. We define the judging problem, the budget-conditioned judge function, the programmatic oracle, and the closed-ended metrics in one place; the open-ended correlation metrics are defined in Appendix H.

E.1 Spaces and Objects

Let $\mathcal{Q}$ be the space of questions, $\mathcal{G}$ the space of ground-truth references, $\mathcal{R}$ the space of student responses, and

$$\mathcal{V} = \{\texttt{Correct}, \texttt{Incorrect}, \texttt{Undefined}\} \quad (3)$$

the verdict space, where `Undefined` denotes an unparseable response. A *benchmark* is a finite set $\mathcal{D} = \{(q_i, a_i^*, r_{s,i})\}_{i=1}^{N_\mathcal{D}}$ of questions q_i, gold answers a_i^*, and student responses $r_{s,i}$ produced by a student model $\mathcal{S}$ (Appendix J). When step-by-step gold reasoning r_i^* is available (the math datasets), the reference passed to the judge is $g_i = r_i^*$; otherwise $g_i = a_i^*$.

E.2 The Judge Function

A judge is a language model $\mathcal{J}$ with parameters $\theta_{\mathcal{J}}$ constrained to $|\theta_{\mathcal{J}}| \leq 14$B. Under token budget $B \in \{10, 8192\}$ it realizes a (stochastic) map

$$f_{\mathcal{J}}^{(B)} : \mathcal{Q} \times \mathcal{G} \times \mathcal{R} \longrightarrow \mathcal{V}, \qquad (q_i, g_i, r_{s,i}) \mapsto \hat{v}_i. \tag{4}$$

Concretely, $f_{\mathcal{J}}^{(B)}$ applies the budget-appropriate prompt template $\mathcal{T}_B$ (Appendix L.1, L.2), tokenizes it through the model's native chat template, decodes a response $y_i \sim \pi_{\mathcal{J}}(\cdot \mid \mathcal{T}_B(q_i, g_i, r_{s,i}))$ of at most B tokens, and parses a verdict $\hat{v}_i = \textsc{Parse}(y_i)$ with the priority cascade of Appendix G. Decoding is deterministic (τ=0) for the quick verdict and for non-thinking models at B=8,192, and uses the recommended sampling configuration (τ=0.6, top-p=0.95, top-k=20, min-p=0) when native thinking is enabled.

E.3 The Programmatic Oracle

Ground-truth verdicts come from a deterministic oracle rather than from human labels, which makes the closed-ended evaluation exactly reproducible. Define a dataset-type-specific equivalence relation

$$\textsc{Equiv}_{\text{type}}(\hat{a}, a^*, \delta) \in \{\top, \bot\}, \tag{5}$$

with tolerance $\delta = 10^{-9}$, where type $\in \{\textsc{numeric}, \textsc{latex}, \textsc{mc}\}$ is selected from the dataset (Appendix G). The oracle verdict for instance i is

$$v_i^* = \begin{cases} \texttt{Correct} & \text{if } \textsc{Equiv}_{\text{type}}(\hat{a}_i, a_i^*, \delta) = \top, \\ \texttt{Incorrect} & \text{otherwise,} \end{cases} \tag{6}$$

where $\hat{a}_i = \textsc{Extract}_{\text{type}}(r_{s,i})$ is the student's final answer recovered by the answer extractor of Appendix F. The judge never sees v_i^*; it must re-derive correctness from $(q_i, g_i, r_{s,i})$ alone.

E.4 Closed-Ended Metrics

Accuracy. Agreement between the judge and the oracle over a configuration is

$$\text{Acc}(\mathcal{J}, B) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\Big[f_{\mathcal{J}}^{(B)}(q_i, g_i, r_{s,i}) = v_i^*\Big], \tag{7}$$

computed over $N = 64{,}824$ judgments (2 students $\times$ 8 datasets) per configuration.

Instruction-Following Rate. The fraction of responses that yield a parseable (non-`Undefined`) verdict is

$$\text{IFR}(\mathcal{J}, B) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\Big[f_{\mathcal{J}}^{(B)}(q_i, g_i, r_{s,i}) \neq \texttt{Undefined}\Big]. \tag{8}$$

Because IFR $> 99.8\%$ for all top judges (Appendix A), Acc and Acc/IFR (accuracy among parsed responses) coincide to within rounding, so we report raw accuracy throughout.

Budget effect. For domain $\mathcal{D}$ we summarize the overthinking effect by the signed difference

$$\Delta(\mathcal{J}, \mathcal{D}) = \text{Acc}(\mathcal{J}, 10\,; \mathcal{D}) - \text{Acc}(\mathcal{J}, 8192\,; \mathcal{D}), \tag{9}$$

where $\Delta > 0$ means quick verdicts win. §4.2 shows that sign Δ is governed by $\mathcal{D}$ (verifiable vs. non-verifiable) rather than by $\mathcal{J}$.

Persona robustness. For a judge evaluated under a set P of persona system prompts (Appendix M), robustness is the range

$$r(\mathcal{J}) = \max_{p \in P} \text{Acc}(\mathcal{J}, p) - \min_{p \in P} \text{Acc}(\mathcal{J}, p), \tag{10}$$

with smaller r indicating a more stable internal correctness boundary.

F Answer Extraction

Before the oracle can score a student response, it must recover the final answer $\hat{a} = \textsc{Extract}_{\text{type}}(r_s)$ from free-form text. Extraction is dataset-type-specific and implemented in `slmjury/parsers/answer.py`. Each extractor is a prioritized cascade: it tries the most reliable surface form first and falls back to looser heuristics, returning the sentinel `UNABLE_TO_EXTRACT` only when every strategy fails (which the oracle then treats as `Incorrect`). Algorithm 1 gives the unified control flow.

Numeric (GSM8K, GSM-Plus). The GSM8K convention places the answer after a #### delimiter. To tolerate repetitive or truncated generations, the extractor collects *all* post-delimiter candidates and returns the modal value, then normalizes it (Appendix G). Fallbacks recover a boxed value, an explicit "the answer is" phrase, a comma-grouped integer such as `12,000`, and finally the last number in the text.

LaTeX (MATH). The primary strategy extracts the content of the last `\boxed{...}` using an explicit brace-depth counter, so nested constructs like `\boxed{\frac{1}{2}}` are recovered intact rather than truncated at the first closing brace.

Algorithm 1 Type-Dispatched Answer Extraction

Require: Student response r_s, dataset type type
Ensure: Extracted answer $\hat{a}$ or `UNABLE_TO_EXTRACT`

```
if r_s is empty then
    return UNABLE_TO_EXTRACT
end if
if type = NUMERIC then
    Try in order: (1) text after the last ####; (2) last \boxed{}; (3) "final answer is" patterns; (4) comma-grouped integers; (5) last number
else if type = LATEX then
    Try in order: (1) last \boxed{} with brace matching; (2) ####; (3) answer phrases; (4) trailing expression
else
    Multiple choice. Try in order: (1) \boxed{X}; (2) "the answer is (X)" patterns; (3) trailing standalone letter; (4) option-with-context; (5) last A–D token; (6) digit 1–4 mapped to A–D
end if
return first non-empty match, else UNABLE_TO_EXTRACT
```

Multiple choice (ARC, HellaSwag, WinoGrande, TruthfulQA). The extractor searches for an explicit option letter, scanning from the end of the response so that a model's final committed choice overrides earlier deliberation. Numeric labels 1–4 are mapped to A–D, and WinoGrande's $\{1, 2\}$ labels to $\{\mathrm{A}, \mathrm{B}\}$, before comparison.

G Answer Normalization and Equivalence

Accurate evaluation requires robust answer comparison across diverse formats. Our equivalence function `answers_are_equivalent()` dispatches to type-specific comparison strategies based on the dataset. This section describes the full normalization pipeline implemented in `slmjury/parsers/normalizer.py`.

G.1 Type-Based Dispatch

The comparison function selects a strategy based on the dataset type. For **numeric** data (GSM8K, GSM-Plus) it normalizes both answers to floating-point and compares with tolerance $\delta = 10^{-9}$; for **LaTeX** (MATH) it applies the multi-strategy pipeline described below; and for **multiple choice** (ARC-Easy, ARC-Challenge, HellaSwag, WinoGrande, TruthfulQA) it extracts the first A/B/C/D/E letter from each answer and compares case-insensitively, mapping WinoGrande's numeric labels (1, 2) to letters (A, B) first.

G.2 Numeric Normalization

The `normalize_numeric()` function handles diverse numeric formats:

1. Remove dollar signs, commas, percent signs, and units (e.g., "kg", "miles").
2. Handle word multipliers: "million" $\rightarrow \times 10^6$, "billion" $\rightarrow \times 10^9$.
3. Strip trailing periods and whitespace.
4. Convert to `float`. Two values agree if their normalized strings match, or if they are numerically close: a and b are equivalent when both are zero, or when the relative error satisfies $|a - b| / \max(|a|, |b|) < \delta$ with $\delta = 10^{-9}$ (an absolute check is used when exactly one value is zero). The relative criterion makes the comparison scale-invariant, so it behaves identically on small and large magnitudes.

G.3 LaTeX Multi-Strategy Comparison

For MATH dataset answers (LaTeX expressions), we employ a five-strategy comparison pipeline. Each strategy is tried in order; the first successful comparison determines the result:

1. **String Normalization**: Normalize both LaTeX strings by removing whitespace, standardizing `\frac`, `\sqrt`, and other LaTeX commands. Compare the normalized strings directly.
2. **Fraction Evaluation**: Handle `\frac{a}{b}` by computing a/b numerically. Also handles `\dfrac` and `\tfrac`. Compare resulting floats with tolerance $\delta = 10^{-9}$.
3. **Symbolic Comparison**: Parse both expressions into SymPy symbolic objects. Use `sympy.simplify(a - b) == 0` to check symbolic equivalence. Handles cases like $x^2 + 2x + 1$ vs. $(x + 1)^2$.
4. **Numeric Evaluation**: Evaluate both SymPy expressions numerically at random points using `sympy.N()`. Compare with tolerance $\delta = 10^{-9}$. Fallback when symbolic simplification times out.

5. **Plain Number Extraction**: Extract plain numbers from LaTeX strings (stripping all formatting) and compare numerically. Last resort for simple numeric answers wrapped in unnecessary LaTeX.

G.4 Verdict Parsing

Judge responses are mapped to a verdict $\hat{v} = \textsc{Parse}(y) \in \mathcal{V}$ by a six-level priority cascade (`slmjury/parsers/judgement.py`). Each level is attempted in turn; the first that matches returns its verdict, and when several keywords occur at one level the *last* match wins, reflecting that a model's final statement is its committed verdict. The levels, in decreasing confidence, are:

1. **Boxed** (high): `\boxed{CORRECT}` or `\boxed{INCORRECT}`, case-insensitive.
2. **Markdown bold** (high): `**CORRECT**` or `**INCORRECT**`.
3. **Labeled** (medium): patterns such as "verdict: ...", "the answer is ...", "my judgement: ...".
4. **Quick verdict** (medium): the response, its first line, or its first word is (or starts with) "correct"/"incorrect". This level handles the B=10 single-word format.
5. **Sentence pattern** (medium): templates like "the solution is ...", "therefore, ...", "I conclude this is ...".
6. **Fallback** (low): the last occurrence of "correct" (negative lookbehind for "in") vs. "incorrect" in the text decides the verdict.

If no level matches, the verdict is `Undefined`, an instruction-following failure that lowers IFR but is excluded from the accuracy numerator. At B=10, levels 4–6 dominate; at B=8,192, the boxed level 1 dominates, so the same parser serves both budgets without modification.

H Open-Ended Scoring Metrics and Parsing

This section formalizes the five correlation metrics used for open-ended evaluation (§3.5) and the score-parsing cascade that turns a free-form scoring response into a number. All metrics are implemented in `slmjury/core/scoring_evaluator.py` and the parser in `slmjury/parsers/score.py`.

H.1 Score Vectors

For a benchmark of n items, let $\mathbf{s} = (s_1, \ldots, s_n)$ be the SLM judge scores and $\mathbf{g} = (g_1, \ldots, g_n)$ the reference scores (human annotations for SummEval, oracle-model scores for MT-Bench), each $s_i, g_i \in \{1, \ldots, 5\}$. Items where the judge response fails to parse are dropped pairwise, leaving $n_{\text{valid}} \leq n$ pairs over which the metrics below are computed.

H.2 Correlation and Agreement Metrics

Pearson r (linear agreement).

$$r(\mathbf{s}, \mathbf{g}) = \frac{\sum_i (s_i - \bar{s})(g_i - \bar{g})}{\sqrt{\sum_i (s_i - \bar{s})^2}\,\sqrt{\sum_i (g_i - \bar{g})^2}}, \quad (11)$$

undefined (reported as $\varnothing$) when either score vector has zero variance.

Spearman ρ (ordinal agreement). Let $\textsc{rk}(\cdot)$ assign fractional (tie-averaged) ranks. Then $\rho = r\big(\textsc{rk}(\mathbf{s}), \textsc{rk}(\mathbf{g})\big)$, i.e. Pearson correlation on ranks, which is robust to monotone nonlinearity in the 1–5 scale.

Cohen's κ (binary agreement). Binarize at the quality threshold $t = 4$ via $b_i = \mathbf{1}[s_i \geq t]$ and $b'_i = \mathbf{1}[g_i \geq t]$. With observed agreement p_o and chance agreement p_e from the 2×2 confusion matrix,

$$\kappa = \frac{p_o - p_e}{1 - p_e}. \quad (12)$$

Thresholded accuracy. The fraction of items on which the binarized labels agree, $\text{Acc} = \frac{1}{n}\sum_i \mathbf{1}[b_i = b'_i]$.

Mean squared error (calibration). $\text{MSE} = \frac{1}{n}\sum_i (s_i - g_i)^2$, capturing absolute scale calibration rather than rank or linear agreement.

For SummEval, these metrics are computed per dimension (coherence, consistency, fluency, relevance) and then averaged; for MT-Bench they are computed over the holistic per-conversation score and aggregated over the four oracle-student combinations.

H.3 Score Parsing

Extracting a numeric score from a judge response is itself a prioritized cascade, because models express scores in many surface forms. Algorithm 2 shows the single-score parser; the SummEval multi-score parser additionally supports structured (`coherence=4, ...`), positional (`3,2,2,3`), and

per-dimension scattered formats, with `\text{}` wrappers and `$. . . $` math mode stripped before matching.

Algorithm 2 Single-Score Parsing (1–5)

Require: Response y, max score $M = 5$
Ensure: Score $s \in \{1, \ldots, M\}$ or `None`
1: **(1)** last `\boxed{N}` or `\boxed{N/M}`
2: **(2)** last markdown `**N**`
3: **(3)** last bracketed `[[N]]`
4: **(4)** labeled "score/rating: N" or "N out of M"
5: **(5)** last standalone integer in $[1, M]$
6: **return** first valid match in range, else `None`

A score of `None` is excluded from the valid pairs above and counted in the parse-failure rate; the strong judges parse cleanly, while sub-1B judges account for most failures (Appendix C).

I Collaborative Judging: Algorithms and Prompts

This section formalizes the two collaborative judging strategies of §3.4: majority-voting ensembles and the Reflect-Critique-Refine (RCR) multi-agent debate protocol. The corresponding result tables are reported in Appendix B.

I.1 Majority-Voting Ensemble

Given three judges producing verdicts $(v_1, v_2, v_3) \in \mathcal{V}^3$ for a problem, the ensemble verdict is the majority label, with ties resolved to `Undefined`:

$$V_{\text{maj}}(v_1, v_2, v_3) = \begin{cases} \texttt{C} & \text{if } \sum_{j=1}^{3} \mathbf{1}[v_j{=}\texttt{C}] \geq 2 \\ \texttt{I} & \text{if } \sum_{j=1}^{3} \mathbf{1}[v_j{=}\texttt{I}] \geq 2 \\ \texttt{U} & \text{otherwise} \end{cases} \tag{13}$$

We draw the panel from the top-5 individual judges (Phi-4@10, Qwen3-14B@10, Qwen3-8B@10, Phi-4-R-Plus@8192, Phi-4-R@8192), each at its best token setting, and enumerate all $\binom{5}{3}$=10 panels. Because the ensemble reuses cached individual judgments, it requires no additional inference.

I.2 RCR Multi-Agent Debate

Algorithm 3 formalizes the RCR procedure. Round 0 uses the standard reasoned-verdict prompt (Appendix L.2); each subsequent round shows every agent its own previous judgement and those of its peers, then asks it to reflect, critique, and refine. Debate stops as soon as the three agents agree (consensus) or after T=5 rounds, in which case the verdict falls back to the majority label of Eq. 13.

Algorithm 3 RCR Multi-Agent Debate for Binary Judging

Require: Query q, agents $\mathcal{A} = \{a_1, \ldots, a_N\}$, max rounds T
Ensure: Consensus verdict v^* and reasoning traces $\mathcal{R}$
1: **Round 0:** Each a_i generates $(v_i^{(0)}, r_i^{(0)}) \sim \pi_{a_i}(\cdot \mid q)$
2: **if** $\forall\, i, j : v_i^{(0)} = v_j^{(0)}$ **then**
3: **return** $(v_i^{(0)}, \{r_i^{(0)}\}_{i=1}^{N})$
4: **end if**
5: **for** $t = 1$ **to** T **do**
6: **for all** $a_i \in \mathcal{A}$ **do**
7: $\mathcal{P}_i^{(t-1)} \leftarrow \{(v_j^{(t-1)}, r_j^{(t-1)})\}_{j \neq i}$
8: **Reflect:** Self-critique c_i^{self} on $r_i^{(t-1)}$
9: **Critique:** Evaluate $\mathcal{P}_i^{(t-1)}$
10: **Refine:** $(v_i^{(t)}, r_i^{(t)}) \leftarrow \text{update}(v_i^{(t-1)}, c_i, \mathcal{P}_i)$
11: **end for**
12: **if** $\forall\, i, j : v_i^{(t)} = v_j^{(t)}$ **then**
13: **return** $(v_i^{(t)}, \bigcup_{i,s \leq t} r_i^{(s)})$
14: **end if**
15: **end for**
16: **return** $(\text{majority}(\{v_i^{(T)}\}), \bigcup_{i,t} r_i^{(t)})$

I.3 Debate Configurations

Table 13 lists all 10 debate configurations. Variant A uses three distinct architectures at τ=0; Variant B uses one model sampled at three temperatures $\tau \in \{0.0, 0.4, 0.9\}$ to inject diversity without architectural change. Three Variant A configurations (marked with †) required a reduced budget of B=4,096 because their multi-round prompts exceeded the context window at B=8,192; these are reported separately in Table 9 and excluded from main-paper claims.

Subprocess isolation. Each agent in a Variant A debate uses a different model. To respect GPU memory limits, agents are loaded in separate subprocesses via Python `multiprocessing` (spawn context): each subprocess loads one vLLM instance, runs inference for all problems in the current round, returns results through a queue, and terminates before the next agent loads. This lets three distinct models debate without requiring si-

Variant	Agent Models	t_1	t_2	t_3
A	LLaMA-3.2-3B, Qwen2.5-1.5B, Qwen3-1.7B	0.0	0.0	0.0
	Qwen2.5-1.5B, Qwen3-1.7B, Phi-4-mini†	0.0	0.0	0.0
	Qwen3-4B, Phi-4-mini, Qwen2.5-3B†	0.0	0.0	0.0
	Qwen3-4B, Qwen2.5-3B, Qwen3-1.7B	0.0	0.0	0.0
	LLaMA-3.2-3B, Qwen3-4B, Phi-4-mini†	0.0	0.0	0.0
B	LLaMA-3.2-1B $\times$ 3	0.0	0.4	0.9
	Qwen2.5-1.5B $\times$ 3	0.0	0.4	0.9
	Qwen3-4B $\times$ 3	0.0	0.4	0.9
	Qwen3-0.6B $\times$ 3	0.0	0.4	0.9
	Phi-4-mini $\times$ 3	0.0	0.4	0.9

Table 13: **MAD debate configurations.** Variant A (cross-architecture): three different models at temperature 0.0. Variant B (intra-model): the same model at three temperatures. Seven configurations ran at `max_tokens=8192`; three (†) ran at `max_tokens=4096` due to context length constraints (see Table 9). All debates run for up to 6 rounds (Round 0 initial + Rounds 1–5 RCR).

multaneous GPU residency.

I.4 RCR Prompt: Math Datasets

Used for GSM8K, GSM-Plus, and MATH.

I.5 RCR Prompt: Science Datasets

Used for ARC-Easy and ARC-Challenge.

I.6 RCR Prompt: General Reasoning Datasets

Used for HellaSwag, WinoGrande, and TruthfulQA. This prompt is domain-neutral and focuses on general reasoning quality.

RCR Debate Prompt: Math

```
You are Agent {agent_id} in a multi-agent debate
to judge whether a student's math answer is
correct.

Question: {question}

Ground truth answer: {ground_truth}

Student answer: {student_answer}

{own_previous}

Here are the judgements from other agents:
{context}

This is debate round {round_num}. Please
carefully analyze all judgements, including your
own, identify any errors in reasoning, and
provide your revised judgement.
- If you believe your previous verdict is correct,
  explain why and defend it.
- If you believe you made an error, explain the
  error and provide a corrected judgement.
- If you believe another agent's verdict is
  correct, explain why you agree with it.

Your final verdict must be:
\boxed{CORRECT} or \boxed{INCORRECT}
```

RCR Debate Prompt: Science

```
You are Agent {agent_id} in a multi-agent debate
to judge whether a student's science answer is
correct.

Question: {question}

Ground truth answer: {ground_truth}

Student answer: {student_answer}

{own_previous}

Here are the judgements from other agents:
{context}

This is debate round {round_num}. Please
carefully analyze all judgements, including your
own, identify any misconceptions or flawed
scientific reasoning, and provide your revised
judgement.
- If you believe your previous verdict is correct,
  explain the scientific principles supporting
  your verdict.
- If you believe you made an error, explain the
  scientific misconception and provide a corrected
  judgement.
- If you believe another agent's verdict is
  correct, explain why their scientific reasoning
  is sound.

Your final verdict must be:
\boxed{CORRECT} or \boxed{INCORRECT}
```

RCR Debate Prompt: General

```
You are Agent {agent_id} in a multi-agent debate
to judge whether a student's answer is correct.

Question: {question}

Ground truth answer: {ground_truth}

Student answer: {student_answer}

{own_previous}

Here are the judgements from other agents:
{context}

This is debate round {round_num}. Please
carefully analyze all judgements, including your
own, identify any errors in reasoning, and
provide your revised judgement.
- If you believe your previous verdict is correct,
  explain clearly why and defend your position.
- If you believe you made an error, explain what
  went wrong and provide a corrected judgement.
- If you believe another agent's verdict is
  correct, explain why you agree with their
  reasoning.

Your final verdict must be:
\boxed{CORRECT} or \boxed{INCORRECT}
```

J Student Solution Generation

This section describes the student solution generation pipeline implemented in `slmjury/core/solver.py`.

J.1 Overview

Two student models generate the solutions that judges evaluate: **Qwen2.5-32B-Instruct-GPTQ-Int8**, a 32B model with GPTQ INT8 quantization, and **LLaMA-3.1-8B-Instruct**, an 8B model; both run on 2 GPUs at 0.85 memory utilization. For each of the 8 closed-ended datasets, both students generate one solution per problem, producing $32{,}412 \times 2 = 64{,}824$ student solutions across all datasets.

J.2 Generation Process

1. **Dataset Loading**: Problems are loaded from the HuggingFace-cached JSON files (Appendix N).

2. **Prompt Construction**: Each problem is formatted using the dataset-type-specific prompt template (Appendix L.3). Science multiple-choice datasets (ARC) use a science-specific prompt, while general multiple-choice datasets (HellaSwag, WinoGrande, TruthfulQA) use a general prompt.

3. **Batch Inference**: vLLM generates solutions using `SamplingParams(temperature=0.7, max_tokens=1024, top_p=0.9)`.

4. **Answer Extraction**: A format-aware parser recovers the final answer: text after the last `####` for numeric, content inside the last `\boxed{}` for LaTeX, and the first A/B/C/D letter for multiple choice.

5. **Result Storage**: Solutions are saved as JSON files with fields: `problem_id`, `question`, `ground_truth_reasoning`, `ground_truth_answer`, `student_reasoning`, `student_answer`, `dataset`, and `model`.

J.3 Student Model Performance

Table 14 summarizes the solver accuracy of both student models across all eight benchmarks. The choice of two student models with different capability levels is deliberate: it ensures that judges are tested on a mix of correct and incorrect solutions, preventing the evaluation from being trivially easy (if all solutions were correct) or trivially hard (if all were incorrect).

J.4 Why Two Student Models?

Using two student models with different accuracy levels serves two purposes. First, it tests whether judges can correctly identify both correct and incorrect solutions. A judge that only evaluates solutions from a near-perfect student would be tested primarily on its ability to confirm correctness, missing the harder task of identifying subtle errors. Second, it reveals whether judge accuracy depends on the quality of the student. If a judge performs well on one student but poorly on another, this indicates that the judge's internal correctness model is not robust to variation in solution style and quality. Our results show that the strongest judges (Phi-4, Qwen3-14B, Qwen3-8B) maintain high accuracy across both students, while weaker judges show larger gaps.

J.5 Oracle Extraction Reliability

Table 15 reports the answer extraction rate (valid responses out of total problems) for each student-dataset pair. The oracle's five-strategy equivalence pipeline (Appendix G) successfully extracts a parseable answer from $\geq$99% of student responses on 13 of 16 student-dataset pairs, with most achieving 100%. The main outlier is MATH under LLaMA-3.1-8B (93.80%, 4,690/5,000), where complex LaTeX formatting and non-standard answer expressions occasionally elude the parser; LLaMA-3.1-8B also shows slightly reduced rates on WinoGrande (98.03%) and ARC-Challenge (99.49%). Responses that fail extraction are excluded from judge evaluation; because these constitute a small fraction of total problems, their exclusion has negligible impact on aggregate judge accuracy metrics.

K Implementation Details

This section documents the hardware, software stack, inference configuration, and compute budget behind all experiments. The formal metric definitions (accuracy, IFR, Δ, persona range) are in Appendix E.

K.1 Hardware

All experiments were conducted on a shared NVIDIA DGX-1 server with 8 $\times$ Tesla V100-SXM2-32GB GPUs, 2 $\times$ Intel Xeon E5-2698 v4 CPUs @ 2.20GHz (40 cores, 80 threads), and

	Math			Science		General			
Student	**GSM8K**	**GSM+**	**MATH**	**ARC-E**	**ARC-C**	**HSwag**	**WGrnd**	**TQA**	**Avg**
Qwen2.5-32B	95.45	76.62	77.40	92.72	88.05	73.71	84.53	69.88	82.30
LLaMA-3.1-8B	82.49	62.69	49.46	77.48	73.29	49.00	61.40	50.58	63.30

Table 14: **Student model accuracy (%) across benchmarks.** Qwen2.5-32B serves as a strong student; LLaMA-3.1-8B as a weaker student. General reasoning tasks (especially TruthfulQA and WinoGrande) are substantially harder for both students.

	Math			Science		General			
Student	**GSM8K**	**GSM+**	**MATH**	**ARC-E**	**ARC-C**	**HSwag**	**WGrnd**	**TQA**	**Avg**
Qwen2.5-32B	100.00	99.74	99.14	99.92	100.00	100.00	100.00	100.00	99.85
LLaMA-3.1-8B	100.00	100.00	93.80	99.71	99.49	99.83	98.03	100.00	98.86

Table 15: **Oracle answer extraction rate (%) per student-dataset pair.** Values show valid/total $\times 100$. Rates exceed 99% for most configurations; the lowest rate is MATH under LLaMA-3.1-8B (93.80%), where complex LaTeX solutions occasionally elude the five-strategy parser.

512GB DDR4 RAM, using up to 2 GPUs per model. Models requiring 1 GPU (0.6B–4B parameter models) run on a single V100. Models requiring 2 GPUs (7B–14B models, student models) use tensor parallelism across 2 V100s. All non-oracle student and judge inference runs locally with vLLM; MT-Bench oracle scoring is the only exception and uses the Together API for GPT-OSS-120B and Qwen3.5-397B-A17B.

K.2 Software Stack

Table 16 lists the exact software versions used across all experiments.

Component	Version
vLLM	0.8.5.post1
Transformers	4.57.1
PyTorch	2.6.0+cu124
SymPy	1.13.1
Python	3.10.18
CUDA	12.4

Table 16: **Software versions** used across all experiments.

K.3 Inference Configuration

All student generation and SLM judge inference use vLLM (Kwon et al., 2023) for high-throughput batched generation. The only non-vLLM inference path is MT-Bench oracle scoring with GPT-OSS-120B and Qwen3.5-397B-A17B through the Together API. Key inference parameters:

- **Quick verdict**: `SamplingParams(max_tokens=10, temperature=0)`
- **Reasoned verdict (non-thinking)**: `SamplingParams(max_tokens=8192, temperature=0)`
- **Reasoned verdict (thinking-enabled)**: `SamplingParams(max_tokens=8192, temperature=0.6, top_p=0.95, top_k=20, min_p=0)`
- **Chat template**: All prompts are tokenized using each model's native chat template via `tokenizer.apply_chat_template()`.
- **Thinking mode**: For Qwen 3 models, `enable_thinking` is passed as a keyword argument to the chat template. When `max_tokens=10`, thinking is disabled.
- **Batch size**: Determined by vLLM's continuous batching scheduler. Models $\leq$4B use `max_num_seqs=64`; Phi-4-Reasoning models use `max_num_seqs=3` due to long output sequences; all others use the default (32).
- **GPU memory**: For local vLLM runs, `gpu_memory_utilization` follows `models.yaml` when specified: student and Phi-4-family runs use 0.85, judge and scoring-judge defaults use 0.8, and student-generation defaults use 0.7.

K.4 Compute Budget

Table 17 provides an estimate of the total compute used.

Experiment	Configurations
Student solutions (closed)	$2 \times 8 = 16$
Individual judging	$13 \times 2 \times 2 \times 8 + 3 \times 1 \times 2 \times 8 = 464$
Persona evaluation	$6 \times 5 \times 2 \times 2 \times 8 = 960$
Ensemble (post-hoc)	10 combos (no inference)
Multi-agent debate	$10 \times 3 \times 5 \times 2 \times 8 = 2{,}400$
SummEval judges	$16 \times 1 = 16$
MT-Bench student solutions	$2 \times 1 = 2$
MT-Bench oracle scoring	$2 \times 2 \times 1 = 4$
MT-Bench judges	$16 \times 2 \times 2 \times 1 = 64$
Total	**>3,900 experiments**

Table 17: **Compute budget breakdown.** Individual: 13 dual-setting judges $\times$ 2 token settings $\times$ 2 students $\times$ 8 datasets + 3 always-thinking judges $\times$ 1 token setting $\times$ 2 students $\times$ 8 datasets = 464. Persona: 6 personas $\times$ 5 judges $\times$ 2 token settings $\times$ 2 students $\times$ 8 datasets. Debate: 10 combos $\times$ 3 agents $\times$ up to 5 rounds $\times$ 2 students $\times$ 8 datasets (worst-case). MT-Bench judges: 16 judges $\times$ 2 oracles $\times$ 2 students. Ensemble voting reuses individual judgments (no additional inference).

L Prompt Templates

This section presents the exact prompt templates used across all experiments. All prompts are reproduced verbatim from the codebase; no modifications were made between experimental runs.

L.1 Judge Prompt: Quick Verdict (10 Tokens)

This prompt is used when `max_tokens=10`. The judge must respond with exactly one word: `Correct` or `Incorrect`.

L.2 Judge Prompt: Reasoned Verdict (8,192 Tokens)

This prompt is used when `max_tokens=8192`. The judge provides step-by-step reasoning followed by a boxed verdict.

L.3 Student Solver Prompts

The student models (Qwen2.5-32B and LLaMA-3.1-8B) generate solutions using dataset-type-specific prompts:

L.4 Open-Ended Scoring Prompts

The following prompts are used for open-ended quality scoring evaluation (§4.5). Both prompts are implemented in `slmjury/core/scoring_judge.py`.

Quick Verdict Prompt (`max_tokens=10`)

```
Your role is to compare the student's answer
to the ground truth and determine correctness.

Respond with exactly one word:
- `Correct': If the student's final answer
  matches the ground truth.
- `Incorrect': If the student's final answer
  does not match.

Focus only on the final answer, ignore
reasoning steps.

Question: {question}

Ground truth answer: {ground_truth}

Student answer: {student_answer}
```

Reasoned Verdict Prompt (`max_tokens=8,192`)

```
Your role is to compare the student's answer
to the ground truth and determine correctness.

First, explain your reasoning for the judgment.
Then, on a new line, give the final verdict in
the exact format:
\boxed{CORRECT} or \boxed{INCORRECT}

Focus only on the final answer when judging
correctness. Ignore reasoning steps in the
student's solution.

Question: {question}

Ground truth answer: {ground_truth}

Student answer: {student_answer}
```

Numeric Dataset Prompt (GSM8K, GSM-Plus)

```
Think step-by-step to solve this math problem.
Give your final answer after ####.

Problem: {question}

Solution:
```

LaTeX Dataset Prompt (MATH)

```
Think step-by-step to solve this math problem.
Put your final answer in \boxed{}.

Problem: {question}

Solution:
```

Science Multiple-Choice Prompt (ARC-Easy, ARC-Challenge)

```
Think step-by-step to answer this science
question. Choose A, B, C, or D.

{question}

Think step-by-step, then give your final answer.
```

General Multiple-Choice Prompt (HellaSwag, WinoGrande, TruthfulQA)

```
Think step-by-step to answer this question.
Choose the best option.

{question}

Think step-by-step, then give your final answer.
```

SummEval Scoring Prompt (4 dimensions, single call)

```
Evaluate the following summary of a news article
on four dimensions.
Rate EACH dimension from 1 (very poor) to 5
(excellent).

Source Article:
{source_text}

Summary to Evaluate:
{summary}

Dimensions:
- Coherence: Is the summary well-organized and
  easy to follow?
- Consistency: Does the summary accurately
  reflect the source article without
  contradictions?
- Fluency: Is the summary grammatically correct
  and well-written?
- Relevance: Does the summary capture the
  important information from the source?

Provide your ratings as:
\boxed{coherence=X, consistency=X, fluency=X,
relevance=X}
```

MT-Bench Scoring Prompt (2-turn holistic)

```
Rate the overall quality of the assistant's
responses in the following 2-turn conversation
on a scale of 1-5.
1 = Very poor, 2 = Poor, 3 = Average,
4 = Good, 5 = Excellent.

[Turn 1 Question]:
{turn1_question}

[Turn 1 Response]:
{turn1_response}

[Turn 2 Question]:
{turn2_question}

[Turn 2 Response]:
{turn2_response}

Consider helpfulness, accuracy, depth,
creativity, and how well the assistant maintains
context across both turns.
Provide your rating as: \boxed{SCORE}
```

M Persona System Prompts

We evaluate each judge under six persona system prompts designed to test robustness to instructed evaluation styles. The prompts below are reproduced **exactly** as used during all experimental runs (from `slmjury/strategies/persona.py`). When a persona is active, the prompt is prepended as a system message before the judge prompt (Appendix L). A seventh "Base" condition uses no system prompt.

Strict Persona

```
You are a highly critical academic reviewer.
Evaluate the answer for factual correctness,
logical consistency, completeness, and absence
of unsupported claims. Penalize vagueness,
missing steps, and incorrect terminology.
Do not reward verbosity unless it improves rigor.
Prefer precise, structured, and well-justified
answers. If uncertain, lower the score.
```

Lenient Persona

```
You are a supportive teacher grading student
work. Focus on whether the core idea is correct
even if wording is imperfect. Give partial credit
for partially correct reasoning. Do not heavily
penalize minor factual or grammatical errors.
Reward effort, clarity of intent, and basic
understanding.
```

Industry Persona

```
You are an industry professional evaluating
usefulness for real-world application. Prioritize
correctness, practicality, and actionability.
Penalize unnecessary theory, long explanations,
and fluff. Prefer short, clear, implementable
answers. An answer that can be used immediately
should score higher.
```

Logic Persona

```
You are a logic-focused evaluator. Judge only
based on internal reasoning quality. Ignore
writing style, tone, and length. Check whether
conclusions follow from premises. Penalize
contradictions, missing steps, and invalid
inferences. Factual correctness matters only if
it affects logical validity.
```

Safety Persona

```
You are a safety auditor checking for
hallucinations and unsupported claims. Penalize
confident statements without evidence. Reward
uncertainty when appropriate. If the answer
invents facts, names, or numbers, assign a low
score. Factual grounding and cautious language
should score higher than fluency.
```

Helpfulness Persona

```
You are judging how helpful the answer is to
the user. Focus on whether the response solves
the user's problem. Reward clarity, directness,
and relevance. Penalize tangents, missing steps,
and unclear instructions. An answer that enables
the user to act should receive a higher score.
```

N Dataset Details

We evaluate SLMJURY on eight closed-ended reasoning benchmarks spanning three domains (mathematical, scientific, and general reasoning), plus two open-ended quality scoring benchmarks. Each closed-ended dataset is loaded from HuggingFace via the `datasets` library and cached locally as JSON for reproducibility. Closed-ended datasets use the **test** split unless noted otherwise. Table 18 summarizes the datasets.

Dataset	Domain	Test size	Answer format
GSM8K	Math	1,319	After ####
GSM-Plus	Math	10,552	Direct numeric
MATH	Math	5,000	\boxed{}
ARC-Easy	Science	2,376	A / B / C / D
ARC-Chall.	Science	1,172	A / B / C / D
HellaSwag	General	10,042	A / B / C / D
WinoGrande	General	1,267	A / B
TruthfulQA	General	684	A / B / C / D
Total (closed-ended)		**32,412**	
SummEval	Summarization	1,600 pairs	1–5 per dimension
MT-Bench	Conversation	80 questions	1–5 overall

Table 18: **Benchmark datasets.** All closed-ended evaluations use the test split. With 2 student models, each judge configuration is evaluated over $32{,}412 \times 2 = 64{,}824$ total closed-ended judgments.

GSM8K (Cobbe et al., 2021) consists of 1,319 grade-school math word problems requiring multi-step arithmetic reasoning. Ground truth answers follow the #### delimiter. Our loader extracts the final numeric answer after this delimiter. Source: `openai/gsm8k`.

GSM-Plus (Li et al., 2024) extends GSM8K with 10,552 adversarially perturbed variants covering numerical substitution, digit expansion, fraction conversion, and other perturbation types. Ground truth is provided as a direct numeric value in the dataset schema. Source: `qintongli/GSM-Plus`.

MATH (Hendrycks et al., 2021) contains 5,000 competition-level mathematics problems across 7 subjects: algebra, counting & probability, geometry, intermediate algebra, number theory, prealgebra, and precalculus. Answers are LaTeX expressions inside `\boxed{}`. Our loader extracts the content of the last `\boxed{}` occurrence using a brace-matching parser that handles nested braces. Source: `EleutherAI/hendrycks_math`.

ARC-Easy and ARC-Challenge (Clark et al., 2018) comprise 2,376 and 1,172 multiple-choice science questions, respectively. Each question has 3–5 answer choices (labeled A, B, C, D, and occasionally E). Our loader formats the question with inline choices: `A) text B) text C) text D) text`, and stores the correct label as ground truth. Source: `allenai/ai2_arc`.

HellaSwag (Zellers et al., 2019) contains 10,042 sentence completion questions requiring commonsense reasoning about everyday activities (**validation** split; no public test labels). Each question presents a context and four possible continuations (A, B, C, D). The correct continuation requires understanding physical and social commonsense, making it substantially harder for judges than factual recall. Our loader formats the context with inline choices. Source: `Rowan/hellaswag`.

WinoGrande (Sakaguchi et al., 2021) provides 1,267 pronoun resolution problems derived from Winograd schemas (**validation** split; no public test labels). Each problem requires choosing between two possible referents (A or B) for an ambiguous pronoun. This benchmark tests commonsense reasoning about physical interactions and social dynamics. Our loader maps numeric labels (1, 2) to letters (A, B). Source: `allenai/winogrande` (`winogrande_xl` config).

TruthfulQA (Lin et al., 2022) consists of 684 multiple-choice questions designed to test truthfulness on topics where popular misconceptions exist (**validation** split; no public test labels). Questions span 38 categories (health, law, conspiracies, etc.) with 4 answer choices per question. Our loader uses the `multiple_choice` config, reading the `choices` list and the `label` field (a ClassLabel index 0–3 mapped to letters A–D). Source: `EleutherAI/truthful_qa_mc`.

SummEval (SLM-Human Correlation) (Fabbri et al., 2021) provides 1,600 machine-generated article-summary pairs annotated by human experts on four dimensions: coherence, consistency, fluency, and relevance (each scored 1–5). We use these human annotations as ground truth to measure SLM-Human agreement: correlations between SLM judge scores and human expert scores reveal how well small models can replicate nuanced human quality judgments on generated text. Source: `mteb/summeval`.

MT-Bench (SLM-LLM Correlation) (Zheng et al., 2023) contains 80 multi-turn conversational questions across eight categories

(coding, extraction, humanities, math, reasoning, roleplay, STEM, writing). We use two large oracle models, GPT-OSS-120B (`openai/gpt-oss-120b`) and Qwen3.5-397B-A17B (`Qwen/Qwen3.5-397B-A17B`), as reference scorers via the Together API, each scoring responses from both student models, yielding four oracle-student combinations per judge. Correlations between SLM judge scores and oracle scores measure SLM-LLM agreement: how well small models can approximate the scoring behavior of much larger models on complex multi-turn interactions. Source: `philschmid/mt-bench`.

Data Schema. All closed-ended datasets are normalized to a unified schema with four fields: `problem_id` (integer), `question` (string), `ground_truth_reasoning` (string, empty for non-math datasets), and `ground_truth_answer` (string). This standardization allows all downstream components (judge inference, evaluation, ensemble voting) to operate uniformly across datasets.

O Model Configurations

Table 19 presents the complete configuration for all 16 judge models and Table 20 presents the 2 student models used in our experiments. All configurations are centralized in `models.yaml` and parsed at runtime.

Thinking Mode Handling. Qwen 3 models support a "thinking" mode activated via the chat template parameter `enable_thinking=True`. When a thinking-capable model receives `max_tokens=10`, we pass `enable_thinking=False` to suppress internal reasoning and force a direct verdict. At `max_tokens=8192`, thinking is enabled, allowing the model to use internal chain-of-thought before producing its verdict. Three models (Phi-4-Reasoning, Phi-4-Reasoning-Plus, Phi-4-mini-Reasoning) always generate thinking tokens regardless of the template flag. These "always-thinks" models are evaluated at 8,192 tokens only and marked with † in Table 2.

P Reproducibility and Licensing

P.1 Code and Data Availability

The complete SLMJURY framework will be released publicly upon acceptance, all evaluation results will be available through an interactive leaderboard, and the full model configuration file (`models.yaml`) with local vLLM parameters is provided in Appendix O.

P.2 Reproducibility Checklist

To reproduce our experiments, the following are required:

1. **Hardware**: An NVIDIA DGX-1 server with 8 × Tesla V100-SXM2-32GB GPUs (our experiments used up to 2 per model). Smaller models (0.6B–4B) can run on a single GPU.

2. **Software**: Python 3.10+, vLLM $\geq$ 0.8.5, Transformers $\geq$ 4.57, SymPy $\geq$ 1.13, PyTorch $\geq$ 2.6 with CUDA 12.4.

3. **Model Access**: All 16 judge models and 2 student models are publicly available on HuggingFace under permissive licenses.

4. **Datasets**: All 8 closed-ended benchmark datasets and both open-ended benchmarks (SummEval, MT-Bench) are publicly available on HuggingFace and are automatically downloaded by the framework.

5. **Determinism**: Non-thinking experiments use `temperature=0.0` for deterministic generation. Thinking-enabled models use `temperature=0.6`, `top_p=0.95`, `top_k=20`, `min_p=0` as recommended by their respective model cards. Multi-agent debate Variant B uses temperatures 0.0, 0.4, and 0.9 as specified.

P.3 Model Licenses

Table 21 lists the license of each model family.

P.4 Dataset Licenses

Table 22 lists the license of each dataset.

Family	Key	HuggingFace Identifier	Params	TP	Max Len	Thinking	Always
LLaMA 3.x	llama3.2-1b	meta-llama/Llama-3.2-1B-Instruct	1B	1	32,768	✗	✗
	llama3.2-3b	meta-llama/Llama-3.2-3B-Instruct	3B	1	40,960	✗	✗
	llama3.1-8b	meta-llama/Llama-3.1-8B-Instruct	8B	2	32,768	✗	✗
Qwen 2.5	qwen2.5-1.5b	Qwen/Qwen2.5-1.5B-Instruct	1.5B	1	32,768	✗	✗
	qwen2.5-3b	Qwen/Qwen2.5-3B-Instruct	3B	1	32,768	✗	✗
	qwen2.5-7b	Qwen/Qwen2.5-7B-Instruct	7B	2	—	✗	✗
Qwen 3	qwen3-0.6b	Qwen/Qwen3-0.6B	0.6B	1	40,960	✓	✗
	qwen3-1.7b	Qwen/Qwen3-1.7B	1.7B	1	40,960	✓	✗
	qwen3-4b	Qwen/Qwen3-4B	4B	1	40,960	✓	✗
	qwen3-8b	Qwen/Qwen3-8B	8B	2	32,768	✓	✗
	qwen3-14b	Qwen/Qwen3-14B	14B	2	32,768	✓	✗
Phi-4	phi4-14b	microsoft/phi-4	14B	2	16,384	✗	✗
	phi4r-14b	microsoft/Phi-4-reasoning	14B	2	32,768	✓	✓
	phi4rp-14b	microsoft/Phi-4-reasoning-plus	14B	2	32,768	✓	✓
	phi4mi-3.8b	microsoft/Phi-4-mini-instruct	3.8B	1	40,960	✗	✗
	phi4mr-3.8b	microsoft/Phi-4-mini-reasoning	3.8B	1	32,768	✓	✓

Table 19: **Judge model configurations.** TP = tensor parallel size (number of GPUs). Max Len = maximum context length passed to vLLM. Thinking = whether the model's chat template includes a <think> block. Always = models that always generate thinking tokens (cannot be disabled); these are evaluated at 8,192 tokens only (marked † in Table 2).

Key	HuggingFace Identifier	Params	TP
qwen2.5-32b	Qwen/Qwen2.5-32B-Instruct-GPTQ-Int8	32B	2
llama3.1-8b	meta-llama/Llama-3.1-8B-Instruct	8B	2

Table 20: **Student model configurations.** The 32B Qwen model uses GPTQ INT8 quantization. Each student model generates solutions for all 8 closed-ended datasets, producing the student answer corpus that judges evaluate.

Model Family	License
LLaMA 3.x	Meta Llama 3.2 Community License
Qwen 2.5	Apache 2.0
Qwen 3	Apache 2.0
Phi-4	MIT

Table 21: **Model licenses.** All models permit research use.

Dataset	License
GSM8K	MIT
GSM-Plus	CC-BY-SA 4.0
MATH	MIT
ARC (Easy/Challenge)	CC-BY-SA 4.0
HellaSwag	MIT
WinoGrande	CC-BY 4.0
TruthfulQA	Apache 2.0
SummEval	MIT
MT-Bench	CC-BY 4.0

Table 22: **Dataset licenses.** All datasets are publicly available for research use.